



Cisco Nexus 9000 Series NX-OS System Management Configuration Guide, Release 10.3(x)

First Published: 2022-08-19

Last Modified: 2024-05-09

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 REFERENCED IN THIS DOCUMENTATION ARE SUBJECT TO CHANGE WITHOUT NOTICE. EXCEPT AS MAY OTHERWISE BE AGREED BY CISCO IN WRITING, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENTATION ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

The Cisco End User License Agreement and any supplemental license terms govern your use of any Cisco software, including this product documentation, and are located at: <https://www.cisco.com/c/en/us/about/legal/cloud-and-software/software-terms.html>. Cisco product warranty information is available at <https://www.cisco.com/c/en/us/products/warranty-listing.html>. US Federal Communications Commission Notices are found here <https://www.cisco.com/c/en/us/products/us-fcc-notice.html>.

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 products and features described herein as in development or available at a future date remain in varying stages of development and will be offered on a when-and if-available basis. Any such product or feature roadmaps are subject to change at the sole discretion of Cisco and Cisco will have no liability for delay in the delivery or failure to deliver any products or feature roadmap items that may be set forth in this document.

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.

The documentation set for this product strives to use bias-free language. For the purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

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)

© 2022–2024 Cisco Systems, Inc. All rights reserved.



CONTENTS

Trademarks ?

PREFACE

[Preface](#) xxvii

[Audience](#) xxvii

[Document Conventions](#) xxvii

[Related Documentation for Cisco Nexus 9000 Series Switches](#) xxviii

[Documentation Feedback](#) xxviii

[Communications, services, and additional information](#) xxviii

[Cisco Bug Search Tool](#) xxix

[Documentation feedback](#) xxix

CHAPTER 1

[New and Changed Information](#) 1

[New and Changed Information](#) 1

CHAPTER 2

[Overview](#) 7

[Licensing Requirements](#) 7

[Supported Platforms](#) 7

[Cisco NX-OS Device Configuration Methods](#) 8

[Configuring with CLI or XML Management Interface](#) 8

[Configuring with Cisco DCNM](#) 9

[Network Time Protocol](#) 9

[Cisco Discovery Protocol](#) 9

[Session Manager](#) 9

[Scheduler](#) 9

[SNMP](#) 10

[Online Diagnostics](#) 10

Onboard Failure Logging	10
SPAN	10
ERSPAN	10
LLDP	10
MPLS Stripping	11
sFlow	11
SMUs	11
Virtual Device Contexts	11
Troubleshooting Features	11

CHAPTER 3

Two-stage Configuration Commit	13
About Two-stage Configuration Commit	13
Guidelines and Limitations	14
Configuring in Two-Stage Configuration Commit Mode	15
Aborting the Two-Stage Configuration Commit Mode	21
Displaying Commit IDs	21
Rollback Capability	22
Viewing Current Session Configurations	22

CHAPTER 4

Configuring Switch Profiles	23
About Switch Profiles	23
Switch Profile Configuration Modes	24
Configuration Synchronization Mode	24
Switch Profile Mode	24
Switch Profile Import Mode	24
Configuration Validation	24
Mutual Exclusion Checks	24
Merge Checks	25
Software Upgrades and Downgrades with Switch Profiles	25
Guidelines and Limitations for Switch Profiles	25
Configuring Switch Profiles	27
Adding or Modifying Switch Profile Commands	29
Importing a Switch Profile	30
Importing Configurations in a vPC Topology	32

Isolating a Peer Switch	33
Deleting a Switch Profile	33
Manually Correcting Mutex and Merge Failures	34
Verifying the Switch Profile Configuration	34
Configuration Examples for Switch Profiles	35
Creating a Switch Profile on a Local and a Peer Switch	35
Verifying the Synchronization Status	38
Showing the Running Configuration	38
Displaying the Switch Profile Synchronization Between the Local and the Peer Switch	39
Displaying Verify and Commit on the Local and the Peer Switch	40
Displaying the Successful and Unsuccessful Synchronization Between the Local and the Peer Switch	41
Displaying the Switch Profile Buffer	41
Importing Configurations	42
Migrating to Cisco NX-OS Release 7.0(3)I2(1) or Higher in a Fabric Extender Straight-Through Topology	44
Replacing a Cisco Nexus 9000 Series Switch	45
Synchronizing Configurations	46
Synchronizing Configurations After a Cisco Nexus 9000 Series Switch Reboots	46
Synchronizing Configurations When the mgmt0 Interface Connectivity Is Lost	46
Reverting an Inadvertent Port Mode Change of Layer 2 to Layer 3 in Global Configuration Mode	46

CHAPTER 5

Configuring Frequency Synchronization 47

About Frequency Synchronization	47
Hybrid SyncE-PTP with External PRC Source	48
Timing Sources	48
Timing Inputs	48
Timing Outputs	49
Timing Source Selection Points	49
Licensing Requirements for Synchronous Ethernet (SyncE)	50
Guidelines and Limitations for Frequency Synchronization	50
Configuring Frequency Synchronization	51
Enabling Frequency Synchronization	51

Configuring Frequency Synchronization on an Interface	52
Verifying the Frequency Synchronization Configuration	54

CHAPTER 6
Precision Time Protocol 59

Functionality	59
PTP Offload	60
Clocks	60
Clocks	61
Process	63
ITU-T Telecom Profiles	64
Telecom Profile G.8275.1	64
Telecom Profile G.8275.2	65
High Availability for PTP	67
Guidelines and limitations for PTP	67
Default PTP Parameters	74
Configure PTP	75
Configure PTP Globally	75
Configure PTP GM	79
Configure PTP on an Interface	80
Configure PTP in Unicast Mode	87
Configure Unicast Mode for IPv4 or IPv6	87
Assign Dynamic Role	88
Assign Master Role	89
Assign Slave Role	90
Configure Unicast Source Address	92
Configure PTP Telecom Profile	92
Configure Global PTP Telecom Profile	92
Configure PTP Telecom Profile 8275-1 on an Interface	95
Configure PTP Telecom Profile 8275-2 on an Interface	98
Profile Defaults	101
Configure PTP Notifications	102
PTP Mixed Mode	105
Configure PTP Interface to Stay in Master State	105
Enable PTP Unicast-Negotiation	106

Enhanced Multicast Scale	109
Timestamp Tagging	109
Configure Timestamp Tagging	110
Configure TTAG Marker Packets and Time Interval	111
Verify PTP Configuration	113
Verify PTP Telecom Profile Configuration	114
Configuration Examples	118
Additional References	120
Related Documents	120

CHAPTER 7

Configuring GPS	121
About GPS	121
Guidelines and Limitations for GPS	121
Configuring GPS Settings for the Grandmaster Clock	122
Verifying the GPS Configuration	123

CHAPTER 8

Configuring GNSS	125
About GNSS	125
Guidelines and Limitations for GNSS	126
Enabling the GNSS Receiver	126
Verifying the GNSS Configuration	128

CHAPTER 9

Configuring NTP	131
About NTP	131
NTP Associations	132
NTP as a Time Server	132
Clock Manager	132
High Availability	132
Virtualization Support	132
Prerequisites for NTP	133
Guidelines and Limitations for NTP	133
Default Settings for NTP	135
Configuring NTP	135
Enabling or Disabling NTP	135

Configuring the Device as an Authoritative NTP Server	136
Configuring an NTP Server and Peer	136
Configuring NTP Authentication	138
Configuring NTP Access Restrictions	140
Configuring the NTP Source IP Address	142
Configuring the NTP Source Interface	142
Configuring NTP Logging	143
Verifying the NTP Configuration	144
Configuration Examples for NTP	144
Additional References	146
Related Documents	146
MIBs	146

CHAPTER 10
Configuring CDP 147

About CDP	147
VTP Feature Support	148
High Availability	148
Virtualization Support	148
Guidelines and Limitations for CDP	148
Default Settings for CDP	149
Configuring CDP	149
Enabling or Disabling CDP Globally	149
Enabling or Disabling CDP on an Interface	150
Configuring Optional CDP Parameters	151
Verifying the CDP Configuration	151
Configuration Example for CDP	152

CHAPTER 11
Configuring System Message Logging 153

About System Message Logging	153
Syslog Servers	154
Secure Syslog Servers	154
Guidelines and Limitations for System Message Logging	154
Default Settings for System Message Logging	155
Configuring System Message Logging	156

Configuring System Message Logging to Terminal Sessions	156
Configuring the Origin ID for Syslog Messages	158
Logging System Messages to a File	159
Configuring Module and Facility Messages Logging	161
Configuring Logging Syslogs Compliant to RFC 5424	163
Configuring Syslog Servers	164
Configuring Secure Syslog Servers	166
Configuring the CA Certificate	167
Enrolling the CA Certificate	167
Configuring Syslog Servers on a UNIX or Linux System	169
Displaying and Clearing Log Files	170
Verifying the System Message Logging Configuration	171
Repeated System Logging Messages	171
Configuration Example for System Message Logging	172
Additional References	172
Related Documents	172

CHAPTER 12

Configuring Smart Call Home	173
About Smart Call Home	173
Smart Call Home - Concepts	174
Destination Profiles	174
Smart Call Home Alert Groups	174
Smart Call Home Message Levels	177
Obtaining Smart Call Home	178
Database Merge Guidelines	179
High Availability	179
Virtualization Support	179
Prerequisites for Smart Call Home	179
Guidelines and Limitations for Smart Call Home	180
Default Settings for Smart Call Home	180
Configuring Smart Call Home	181
Configuring Contact Information	181
Creating a Destination Profile	183
Modifying a Destination Profile	184

Associating an Alert Group with a Destination Profile	186
Adding Show Commands to an Alert Group	187
Configuring the Email Server	188
Configuring VRFs To Send Messages Using HTTP	190
Configuring an HTTP Proxy Server	191
Configuring Periodic Inventory Notifications	192
Disabling Duplicate Message Throttling	193
Enabling or Disabling Smart Call Home	193
Configuring SMTP-AUTH for Call Home Mail Transfer	194
Configuring a Source Interface	197
Testing the Smart Call Home Configuration	197
Verifying the Smart Call Home Configuration	198
Configuration Examples for Smart Call Home	198
Additional References	200
Event Triggers	200
Message Formats	201
Short Text Message Format	201
Common Event Message Fields	202
Alert Group Message Fields	204
Fields for Reactive and Proactive Event Messages	204
Fields for Inventory Event Messages	204
Fields for User-Generated Test Messages	205
Sample Syslog Alert Notification in Full-Text Format	205
Sample Syslog Alert Notification in XML Format	208
MIBs	211

CHAPTER 13
Configuring Session Manager 213

About Session Manager	213
High Availability	214
Prerequisites for Session Manager	214
Guidelines and Limitations for Session Manager	214
Configuring Session Manager	214
Creating a Session	214
Configuring ACLs in a Session	215

Verifying a Session	216
Committing a Session	216
Saving a Session	216
Discarding a Session	217
Verifying the Session Manager Configuration	217
Configuration Example for Session Manager	217
Additional References	218
Related Documents	218

CHAPTER 14

Configuring the Scheduler	219
About the Scheduler	219
Remote User Authentication	220
Logs	220
High Availability	220
Prerequisites for the Scheduler	220
Guidelines and Limitations for the Scheduler	220
Default Settings for the Scheduler	221
Configuring the Scheduler	221
Enabling or Disabling the Scheduler	221
Defining the Scheduler Log File Size	222
Configuring Remote User Authentication	222
Defining a Job	223
Deleting a Job	224
Defining a Timetable	225
Clearing the Scheduler Log File	227
Verifying the Scheduler Configuration	227
Configuration Examples for the Scheduler	228
Creating a Scheduler Job	228
Scheduling a Scheduler Job	228
Displaying the Job Schedule	228
Displaying the Results of Running Scheduler Jobs	228

CHAPTER 15

Simple Network Management Protocol	231
SNMP Fundamentals	231

SNMP Functional Overview	235
SNMP Notifications	236
SNMPv3	237
Security Models and Levels for SNMPv1, v2, v3	238
User-Based Security Model	239
CLI and SNMP User Synchronization	240
Group-Based SNMP Access	242
SNMP and Embedded Event Manager	243
Multiple Instance Support	243
High Availability for SNMP	243
Virtualization Support for SNMP	243
Guidelines and Limitations	243
Default Settings for SNMP	245
Configure SNMP	246
Configure SNMP Users	246
Generate Hashed Password Offline	248
Enforce SNMP Message Encryption	248
Assign SNMPv3 Users to Multiple Roles	249
Create SNMP Communities	249
Filter SNMP Requests	250
Configure SNMP Notification Receivers	251
Configure Source Interface for SNMP Notifications	252
Configure Notification Target User	253
Configure SNMP Notification Receivers with VRFs	254
Configure SNMP to Send Traps Using an Inband Port	255
Enable SNMP Notifications	256
Disable Link Notifications on an Interface	264
Display SNMP ifIndex for an Interface	265
Enable One-Time Authentication for SNMP over TCP	265
Assign SNMP Device Contact and Location Information	266
Configure Context to Network Entity Mapping	266
Disable SNMP	267
Modify SNMP Server Counter Cache Update Timer	268
Modify AAA Synchronization Time	269

Configure SNMP Local Engine ID	269
Verify SNMP Configuration	270
Configuration Examples	271
Additional References	273
Related Documents	273
RFCs	273
MIBs	273

CHAPTER 16

Configuring RMON	275
About RMON	275
RMON Alarms	275
RMON Events	276
High Availability for RMON	276
Virtualization Support for RMON	276
Guidelines and Limitations for RMON	277
Default Settings for RMON	277
Configuring RMON	277
Configuring RMON Alarms	277
Configuring RMON Events	279
Verifying the RMON Configuration	279
Configuration Examples for RMON	280
Additional References	280
MIBs	280

CHAPTER 17

Configuring Online Diagnostics	281
About Online Diagnostics	281
Bootup Diagnostics	281
Runtime or Health Monitoring Diagnostics	282
On-Demand Diagnostics	287
High Availability	288
Virtualization Support	288
Guidelines and Limitations for Online Diagnostics	288
Default Settings for Online Diagnostics	289
Configuring Online Diagnostics	289

Setting the Bootup Diagnostic Level	289
Activating a Diagnostic Test	290
Starting or Stopping an On-Demand Diagnostic Test	291
Simulating Diagnostic Results	292
Clearing Diagnostic Results	293
Verifying the Online Diagnostics Configuration	293
Configuration Examples for Online Diagnostics	294

CHAPTER 18 **Configuring the Embedded Event Manager** 295

About EEM	295
Policies	295
Event Statements	296
Action Statements	297
VSH Script Policies	298
Environment Variables	298
EEM Event Correlation	299
High Availability	299
Virtualization Support	299
Prerequisites for EEM	299
Guidelines and Limitations for EEM	299
Default Settings for EEM	300
Configuring EEM	301
Defining an Environment Variable	301
Defining a User Policy Using the CLI	301
Configuring Event Statements	302
Configuring Action Statements	307
Defining a Policy Using a VSH Script	309
Registering and Activating a VSH Script Policy	309
Overriding a Policy	310
Configuring Memory Thresholds	311
Configuring Syslog as EEM Publisher	313
Verifying the EEM Configuration	314
Configuration Examples for EEM	315
Event Log Auto-Collection and Backup	316

Extended Log File Retention	316
Enabling Extended Log File Retention For All Services	316
Disabling Extended Log File Retention For All Services	317
Enabling Extended Log File Retention For a Single Service	317
Displaying Extended Log Files	318
Displaying Global Dictionary Per Log Statistics	319
Disabling Extended Log File Retention For a Single Service	320
Trigger-Based Event Log Auto-Collection	321
Enabling Trigger-Based Log File Auto-Collection	321
Log-Profile YAML File	321
Auto-Collection YAML File	322
Limiting the Amount of Auto-Collections Per Component	327
Auto-Collection Log Files	328
Verifying Trigger-Based Log Collection	331
Checking Trigger-Based Log File Generation	331
Local Log File Storage	332
Generating a Local Copy of Recent Log Files	332
External Log File Storage	334

CHAPTER 19

Configuring MAC Move Policy	337
About MAC Move Policy	337
Guidelines and Limitations for MAC Move Policy	337
Configuring MAC Move Policy	338
Verifying the MAC Move Policy Configuration	339

CHAPTER 20

Terminal Lock for VSH Sessions	341
Terminal Lock for VSH Sessions	341

CHAPTER 21

Configuring Onboard Failure Logging	345
About OBFL	345
Prerequisites for OBFL	346
Guidelines and Limitations for OBFL	346
Default Settings for OBFL	346
Configuring OBFL	346

Verifying the OBFL Configuration	349
Configuration Example for OBFL	350
Additional References	350
Related Documents	350

CHAPTER 22

Switched Port Analyzer 351

About SPAN	351
Ports	351
Characteristics of Source Ports	353
SPAN Destination Ports	353
Characteristics of Destination Ports	354
Sessions	354
Localized SPAN Sessions	354
SPAN Functionality	355
ACL TCAM Regions	355
High Availability	355
Prerequisites	355
Guidelines and Limitations for SPAN	356
SPAN Limitations for the Cisco Nexus 3000 Platform Switches	359
SPAN Limitations for the Cisco Nexus 9200 Platform Switches (excluding 9232E-B1)	360
SPAN Limitations for the Cisco Nexus 9300 Platform Switches	360
SPAN Limitations for the Cisco Nexus 9500 Platform Switches	363
SPAN Guidelines and Limitations for Cisco Nexus 9800 and 9232E-B1 Platform Switches	365
Default Settings	365
Configuring SPAN	366
Configuring a SPAN Session	366
Configuring UDF-Based SPAN	369
Configuring SPAN Truncation	372
Configuring SPAN for Multicast Tx Traffic Across Different LSE Slices	373
Configuring SPAN to CPU	374
Introduction	374
Guidelines and Limitations	374
Configuring SPAN to CPU	375
Shutting Down or Resuming a SPAN Session	376

Verifying the SPAN Configuration	377
Configuration Examples for SPAN	378
Configuration Example for a SPAN Session	378
Configuration Example for a Unidirectional SPAN Session	379
Configuration Example for a SPAN ACL	379
Configuration Examples for UDF-Based SPAN	380
Configuration Example for SPAN Truncation	380
Configuration Examples for Multicast Tx SPAN Across LSE Slices	381
Additional References	382
Related Documents	382

CHAPTER 23

Configuring ERSPAN	383
About ERSPAN	383
ERSPAN Sources	383
ERSPAN Destination	384
ERSPAN Sessions	384
Localized ERSPAN Sessions	384
ERSPAN Truncation	384
Prerequisites for ERSPAN	385
Guidelines and Limitations for ERSPAN	385
Default Settings	389
Configuring ERSPAN	390
Configuring an ERSPAN Source Session	390
Shutting Down or Activating an ERSPAN Session	393
Configuring an ERSPAN ACL	394
Verifying ERSPAN ACL Configuration	397
Configuring UDF-Based ERSPAN	397
Configuring ERSPAN Truncation	400
Configuring an ERSPAN Destination Session	401
Verifying the ERSPAN Configuration	403
Configuration Examples for ERSPAN	404
Configuration Example for an ERSPAN Source Session Over IPv6	404
Configuration Example for a Unidirectional ERSPAN Session	404
Configuration Example for an ERSPAN ACL	404

Configuration Example for a Marker Packet	405
Configuration Examples for UDF-Based ERSPAN	406
Configuration Example for ERSPAN Truncation	406
Configuration Example for an ERSPAN Destination Session Over IPv4	407
Configuration Example for an ERSPAN Destination Session Over IPv6	407

CHAPTER 24

Configuring LLDP 409

About LLDP	409
About DCBXP	410
High Availability	411
Virtualization Support	411
Guidelines and Limitations for LLDP	411
Default Settings for LLDP	412
Configuring LLDP	413
Enabling or Disabling LLDP Globally	413
Enabling or Disabling LLDP on an Interface	414
Configuring DCBXP Egress Queuing	415
Configuring the DCBXP Protocol Version	415
Multiple LLDP Neighbors Per Physical Interface	416
Enabling or Disabling LLDP Multi-Neighbor Support	417
Enabling or Disabling LLDP Support on Port-Channel Interfaces	419
Configuring Optional LLDP Parameters	421
Verifying the LLDP Configuration	422
Configuration Example for LLDP	423

CHAPTER 25

Configuring NetFlow 425

About NetFlow	425
Dual-Layer NetFlow Implementation	426
Flow Records	426
Flow Exporters	426
Export Format	427
Layer 2 NetFlow Keys	427
Flow Monitors	427
NetFlow Output Interface	427

High Availability	428
Prerequisites for NetFlow	428
Guidelines and Limitations for NetFlow	428
Configure Ingress NetFlow	432
Enable NetFlow feature	433
Create a flow record	433
Specify match parameters	434
Specify collect parameters	435
Create a flow exporter	436
Create Flow Monitor	437
Apply Flow Monitor to an Interface	438
Configure Bridged NetFlow on VLAN	439
Configure Layer 2 NetFlow Keys	440
Configure Layer 3 NetFlow on Layer 2 Interfaces	441
Configure NetFlow Timeouts	442
Ingress NetFlow Verification Commands	442
Monitoring NetFlow	444
Display Example for NetFlow	444
Configuration Example for NetFlow	444

CHAPTER 26

Configuring Mixed Mode	445
About Mixed Mode	445
Guidelines and Limitations for Mixed Mode	445
Mixed Mode: Use Cases	446
Use Case: Switches Already Deployed with Feature Analytics	447
Use Case: Switches Already Deployed with Feature NetFlow	447
Use Case: Switches that have Neither Features Configured	448
Verifying the Mixed Mode Configuration	449
Display Example for Mixed Mode	450

CHAPTER 27

Configuring sFlow	453
sFlow	453
sFlow Agent	453
Prerequisites for sFlow	454

Guidelines and Limitations for sFlow	454
Default Settings for sFlow	456
Configuring sFlow	457
Enabling sFlow	457
Configuring the Sampling Rate	457
Configuring the Maximum Sampled Size	458
Configuring the Counter Poll Interval	459
Configuring the Maximum Datagram Size	460
Configuring the sFlow Collector Address	460
Configuring the sFlow Collector Port	461
Configuring the sFlow Agent Address	462
Configuring the sFlow Sampling Data Source	463
Configuring sFlow Extended BGP (Gateway)	464
Verifying the sFlow Configuration	465
Monitoring and Clearing sFlow Statistics	465
Configuration Examples for sFlow	465
Additional References	466
Related Documents	466

CHAPTER 28

Configuring TAP Aggregation and MPLS Stripping	467
About TAP Aggregation	467
Network TAPs	467
TAP Aggregation	468
Guidelines and Limitations for TAP Aggregation	468
About MPLS Stripping	470
Guidelines and Limitations for MPLS Stripping	471
Configuring TAP Aggregation	472
Enabling TAP Aggregation for Line Cards	472
Configuring a TAP Aggregation Policy	472
Attaching a TAP Aggregation Policy to an Interface	474
Configuring Selective Q-in-Q with Provider VLANs	475
Verifying the TAP Aggregation Configuration	476
Configuration Example for TAP Aggregation	477
Configuring MPLS Stripping	477

Enabling MPLS Stripping	477
Configuring the Incoming Port for the VLAN Tag	478
Adding and Deleting MPLS Labels	479
Configuring Destination MAC Addresses	480
Configuring MPLS Label Aging	481
Clearing MPLS Stripping Counters and Label Entries	481
Configuration Examples for MPLS Stripping	482
Additional References	482
Related Documents	482

CHAPTER 29

Configuring MPLS Access Lists 485

Configuring MPLS Access Lists	485
Verifying the MPLS Access Lists Configuration	486
Configuration Examples for MPLS Access Lists	486

CHAPTER 30

Configuring Header Stripping Features for Nexus Data Broker 487

Introduction to Header Stripping Features for Nexus Data Broker	487
Guidelines and Limitations for Header Stripping	489
VXLAN and iVXLAN Header Stripping for Nexus Data Broker	490
About Nexus Data Broker – VXLAN and iVXLAN Header Stripping	490
Supported PIDs to Strip VXLAN and iVXLAN	490
Guidelines and Limitations for VXLAN and iVXLAN Header Strip	490
Configuring Nexus Data Broker Termination	491
Configuration Example for VXLAN and iVXLAN Header Strip	494
ERSPAN Header Stripping for Nexus Data Broker	494
About ERSPAN Header Stripping	494
Supported PIDs to Strip the ERSPAN Header	495
Guidelines and Limitations for ERSPAN Header Stripping	495
Configuring ERSPAN Header Stripping	495
Configuration Example for ERSPAN Header Stripping	497
Verifying the Configuration for ERSPAN Header Stripping	497
GRE Header Stripping for Nexus Data Broker	498
About NDB GRE Header Stripping	498
NDB GRE Header Stripping Guidelines and Limitations	498

CLIs for GRE Header Strip Feature	499
Configuration for Egress and Ingress Ports	499
MPLS Header Stripping for Nexus Data Broker	500
About NDB MPLS Header Stripping	500
NDB MPLS Header Stripping Guidelines and Limitations	501
Commands for MPLS Header Strip Feature	503
Configuration for Egress and Ingress Ports	503

CHAPTER 31 **Configuring Graceful Insertion and Removal** 505

About Graceful Insertion and Removal	505
Profiles	506
Snapshots	507
Guidelines and Limitations for GIR	507
GIR Workflow	508
Configuring the Maintenance-Mode Profile	509
Configuring the Normal-Mode Profile	510
Creating a Snapshot	512
Adding Show Commands to Snapshots	513
Triggering Graceful Removal	514
Triggering Graceful Insertion	517
Maintenance Mode Enhancements	519
Verifying the GIR Configuration	520
Configuration Examples for GIR	521

CHAPTER 32 **Performing Software Maintenance Upgrades** 523

About SMUs	523
Package Management	524
Impact of Package Activation and Deactivation	525
Prerequisites for SMUs	525
Guidelines and Limitations for SMUs	525
Performing a Software Maintenance Upgrade for Cisco NX-OS	526
Preparing for Package Installation	526
Downloading the SMU Package File from Cisco.com	527
Copying the Package File to a Local Storage Device or Network Server	528

Adding and Activating Packages	531
Committing the Active Package Set	533
Deactivating and Removing Packages	534
No-Reload Options for SMU Installation	535
Advanced SMU Installation Methods	541
Installing Multiple SMU Packages Using a Single TAR File	541
Installing SMU Packages as Part of the New NX-OS Software Image Installation	541
Downgrading Feature RPMs	543
Displaying Installation Log Information	544
Performing a Software Maintenance Upgrade for Guest Shell Bash	545
Additional References	546
Related Documents	546

CHAPTER 33

Performing Configuration Replace 547

About Configuration Replace and Commit-timeout	547
Overview	547
Benefits of Configuration Replace	549
Guidelines and Limitations for Configuration Replace	549
Recommended Workflow for Configuration Replace	551
Performing a Configuration Replace	552
Verifying Configuration Replace	554
Examples for Configuration Replace	555

CHAPTER 34

Configuring Rollback 561

About Rollbacks	561
Automatically Generated System Checkpoints	562
High Availability	562
Virtualization Support	562
Prerequisites for Rollbacks	562
Guidelines and Limitations for Rollbacks	562
Default Settings for Rollbacks	563
Configuring Rollbacks	563
Creating a Checkpoint	564
Implementing a Rollback	564

Verifying the Rollback Configuration 565

Configuration Example for Rollback 565

Additional References 566

Related Documents 566

CHAPTER 35

Integrity Check of Candidate Config 567

About Candidate Config 567

Guidelines and Limitations for Candidate Config Integrity Check 567

Performing Integrity Check for Candidate Config 568

Examples of Integrity Check 568

CHAPTER 36

Performing Secure Erase 571

Information about Secure Erase 571

Prerequisites for Performing Secure Erase 571

Guidelines and Limitations for Secure Erase 572

Configuring Secure Erase 572

APPENDIX A

IETF RFCs supported by Cisco NX-OS System Management 581

IETF RFCs Supported by Cisco NX-OS System Management 581

APPENDIX B

Embedded Event Manager System Events and Configuration Examples 583

EEM System Policies 583

EEM Events 586

Configuration Examples for EEM Policies 587

Configuration Examples for CLI Events 587

Monitoring Interface Shutdown 587

Monitoring Module Powerdown 587

Adding a Trigger to Initiate a Rollback 588

Configuration Examples to Override (Disable) Major Thresholds 588

Preventing a Shutdown When Reaching a Major Threshold 588

Disabling One Bad Sensor 588

Disabling Multiple Bad Sensors 588

Overriding (Disabling) an Entire Module 589

Overriding (Disabling) Multiple Modules and Sensors 589

Enabling One Sensor While Disabling All Remaining Sensors of All Modules	590
Enabling Multiple Sensors While Disabling All Remaining Sensors of All Modules	590
Enabling All Sensors of One Module While Disabling All Sensors of the Remaining Modules	590
Enabling a Combination of Sensors on Modules While Disabling All Sensors of the Remaining Modules	591
Configuration Examples to Override (Disable) Shutdown for Fan Tray Removal	591
Overriding (Disabling) a Shutdown for Removal of One or More Fan Trays	591
Overriding (Disabling) a Shutdown for Removal of a Specified Fan Tray	591
Overriding (Disabling) a Shutdown for Removal of Multiple Specified Fan Trays	592
Overriding (Disabling) a Shutdown for Removal of All Fan Trays Except One	592
Overriding (Disabling) a Shutdown for Removal of Fan Trays Except for a Specified Set of Fan Trays	592
Overriding (Disabling) a Shutdown for Removal of All Fan Trays Except One from a Set of Fan Trays	592
Configuration Examples to Create a Supplemental Policy	593
Creating a Supplemental Policy for the Fan Tray Absent Event	593
Creating a Supplemental Policy for the Temperature Threshold Event	593
Configuration Examples for the Power Over-Budget Policy	593
Shutting Down Modules	594
Shutting Down a Specified List of Modules	594
Configuration Examples to Select Modules to Shut Down	594
Using the Policy Default to Select Nonoverridden Modules to Shut Down	594
Using Parameter Substitution to Select Nonoverridden Modules to Shut Down	594
Configuration Examples for the Online Insertion Removal Event	595
Configuration Example to Generate a User Syslog	595
Configuration Example to Monitor Syslog Messages	595
Configuration Examples for SNMP Notification	596
Polling an SNMP OID to Generate an EEM Event	596
Sending an SNMP Notification in Response to an Event in the Event Policy	596
Configuration Example for Port Tracking	596
Configuration Example to Register an EEM Policy with the EEM	597

APPENDIX C

Configuration Limits for Cisco NX-OS System Management	601
Configuration Limits for Cisco NX-OS System Management	601



Preface

This preface includes the following sections:

- [Audience, on page xxvii](#)
- [Document Conventions, on page xxvii](#)
- [Related Documentation for Cisco Nexus 9000 Series Switches, on page xxviii](#)
- [Documentation Feedback, on page xxviii](#)
- [Communications, services, and additional information, on page xxviii](#)

Audience

This publication is for network administrators who install, configure, and maintain Cisco Nexus switches.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
<i>Italic</i>	Italic text indicates arguments for which you supply the values.
[x]	Square brackets enclose an optional element (keyword or argument).
[x y]	Square brackets enclosing keywords or arguments that are separated by a vertical bar indicate an optional choice.
{x y}	Braces enclosing keywords or arguments that are separated by a vertical bar indicate a required choice.
[x {y z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.

Convention	Description
<code>variable</code>	Indicates a variable for which you supply values, in context where italics cannot be used.
<code>string</code>	A nonquoted set of characters. Do not use quotation marks around the string or the string includes the quotation marks.

Examples use the following conventions:

Convention	Description
<code>screen font</code>	Terminal sessions and information the switch displays are in screen font.
<code>boldface screen font</code>	Information that you must enter is in boldface screen font.
<i><code>italic screen font</code></i>	Arguments for which you supply values are in italic screen font.
<code><></code>	Nonprinting characters, such as passwords, are in angle brackets.
<code>[]</code>	Default responses to system prompts are in square brackets.
<code>!, #</code>	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Related Documentation for Cisco Nexus 9000 Series Switches

The entire Cisco Nexus 9000 Series switch documentation set is available at the following URL:

https://www.cisco.com/en/US/products/ps13386/tsd_products_support_series_home.html

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus9k-docfeedback@cisco.com. We appreciate your feedback.

Communications, services, and additional information

- To receive timely, relevant information from Cisco, sign up at [Cisco Profile Manager](#).
- To get the business impact you're looking for with the technologies that matter, visit [Cisco Services](#).
- To submit a service request, visit [Cisco Support](#).
- To discover and browse secure, validated enterprise-class apps, products, solutions, and services, visit [Cisco DevNet](#).
- To obtain general networking, training, and certification titles, visit [Cisco Press](#).
- To find warranty information for a specific product or product family, access [Cisco Warranty Finder](#).

Cisco Bug Search Tool

[Cisco Bug Search Tool](#) (BST) is a gateway to the Cisco bug-tracking system, which maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. The BST provides you with detailed defect information about your products and software.

Documentation feedback

To provide feedback about Cisco technical documentation, use the feedback form available in the right pane of every online document.



CHAPTER 1

New and Changed Information

- [New and Changed Information](#), on page 1

New and Changed Information

Table 1: New and Changed Features

Feature	Description	Changed in Release	Where Documented
Flexible PTP over IPv6 multicast	Added new global and interface commands to set the transmit and receive scope for PTP over IPv6 multicast configuration	10.3(5)M	Configure PTP Globally , on page 75 Configure PTP on an Interface , on page 80
Enhancement to completely support Syslog Protocol (RFC 5424)	Enhanced support for complete compliance with the RFC 5424 standard for Syslog Protocol	10.3(4a)M	Guidelines and Limitations for System Message Logging , on page 154
NTP Security Enhancements - RFC 8573	Enhanced NTP security with the AES128CMAC authentication mechanism along with Type-6 encryption support for authentication keys.	10.3(3)F	Guidelines and Limitations for NTP , on page 133 Configuring NTP Authentication , on page 138 Configuration Examples for NTP , on page 144

Feature	Description	Changed in Release	Where Documented
Multiple followers' support for G8275.2 in Unicast mode	Provided support for G.8275.2 Telecom profile dynamic PTP unicast port, multiple unicast follower port support on Cisco Nexus 9300-FX3 platform switches.	10.3(3)F	Telecom Profile G.8275.2, on page 65 Guidelines and limitations for PTP, on page 67 Configure PTP Globally, on page 75 Configure PTP on an Interface, on page 80 Assign Dynamic Role, on page 88 Configure Unicast Source Address, on page 92 Configure PTP Telecom Profile, on page 92 Profile Defaults, on page 101 Verify PTP Configuration, on page 113 Verify PTP Telecom Profile Configuration, on page 114
Logging 2.0 enhancements	Added support for default bloggerd auto-collect for adjmgr, cts, l2fm, and vmtracker.	10.3(3)F	Guidelines and Limitations for EEM, on page 299
Extended VRF-ID in NetFlow support on all Cisco Nexus 9000 switches	Added support for Ingress_VRF_ID support for NetFlow and Analytics features on all Cisco Nexus 9000 switches.	10.3(3)F	Guidelines and Limitations for Mixed Mode, on page 445 Verifying the Mixed Mode Configuration, on page 449 Display Example for Mixed Mode, on page 450

Feature	Description	Changed in Release	Where Documented
Configuration Replace	Support is added for configuration replace configuration.	10.3(3)F	Guidelines and Limitations for Configuration Replace, on page 549 Recommended Workflow for Configuration Replace, on page 551 Performing a Configuration Replace, on page 552
Expanded support for Type-6 password encryption	Added Type-6 encryption support for SNMPv3 user password.	10.3(3)F	Guidelines and Limitations, on page 243 Configure SNMP Users, on page 246
Support for Source Interface for call home	Added support for source interface for call home	10.3(2)F	Guidelines and Limitations for Smart Call Home, on page 180 Configuring Smart Call Home, on page 181 Configuring a Source Interface, on page 197
GM/GPS/GNSS support on N9K-C93180YC-FX3	GM, GPS, and GNSS functionality is supported on Cisco Nexus N9K-C93180YC-FX3 platform switches.	10.3(2)F	Clocks, on page 61 Guidelines and limitations for PTP, on page 67 Configure PTP Globally, on page 75 Configure PTP GM, on page 79 Configuring GPS, on page 121 Configuring GNSS, on page 125

Feature	Description	Changed in Release	Where Documented
VPLS Stripping	Added support for EoMPLS header stripping on Cisco Nexus 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, and 9300-GX2 TORs.	10.3(2)F	Guidelines and Limitations for MPLS Stripping, on page 471 About NDB MPLS Header Stripping, on page 500 NDB MPLS Header Stripping Guidelines and Limitations, on page 501
Additional LLDP TLVs for ecosystem integrations	Added 802.1 link aggregation, 802.1 vlan name, and 802.3 max frame size optional parameters to the existing lldp tlv-select command	10.3(2)F	Guidelines and Limitations for LLDP , on page 411 Configuring Optional LLDP Parameters, on page 421
PTP Media profile support with Cisco Nexus 9408 platform	PTP on media profile is supported on Cisco Nexus 9408 platform switches.	10.3(2)F	Guidelines and limitations for PTP, on page 67
Flow visibility in Nexus Dashboard Insights and NetFlow	Added support for Mixed Mode	10.3(1)F	Configuring Mixed Mode, on page 445
Support enhanced maintenance mode	During an unexpected reload, ensure switching to maintenance mode by delaying interface UP until the configuration is merged with the mmode profile.	10.3(1)F	Guidelines and Limitations for GIR, on page 507 Triggering Graceful Removal, on page 514 Adding Show Commands to Snapshots, on page 513
IPv6 destination support for sFlow	Added support for IPv6 collector. Note that the source ip address and the collector ip address must belong to the same address family.	10.3(1)F	Guidelines and Limitations for sFlow, on page 454 Configuring the sFlow Collector Address, on page 460 Configuring the sFlow Agent Address , on page 462 Configuration Examples for sFlow, on page 465

Feature	Description	Changed in Release	Where Documented
ERSPAN	Added support for ERSPAN on Cisco Nexus 9808 platform switches.	10.3(1)F	Guidelines and Limitations for ERSPAN Configuring ERSPAN Truncation , on page 400 Verifying the ERSPAN Configuration , on page 403
CDP	Added support for CDP on Cisco Nexus 9808 platform switches.	10.3(1)F	Guidelines and Limitations for CDP , on page 148
LLDP	Added support for LLDP on Cisco Nexus 9808 platform switches.	10.3(1)F	Guidelines and Limitations for LLDP , on page 411
SNMP	Added support for SNMP on Cisco Nexus 9808 platform switches.	10.3(1)F	Guidelines and Limitations , on page 243
PTP	Added support for PTP Media Profile and One-Step Mode on Cisco Nexus 9808 platform switches.	10.3(1)F	Guidelines and limitations for PTP , on page 67
sFlow	Added support for sFlow on Cisco Nexus 9808 platform switches.	10.3(1)F	Guidelines and Limitations for sFlow , on page 454
SPAN	Added support for SPAN on Cisco Nexus 9808 platform switches.	10.3(1)F	Guidelines and Limitations for SPAN , on page 356 SPAN Guidelines and Limitations for Cisco Nexus 9800 and 9332E-B1 Platform Switches , on page 365 Configuring SPAN Truncation , on page 372 Verifying the SPAN Configuration , on page 377
Generic Online Diagnostics (GOLD)	Added support for GOLD on Cisco Nexus 9808 platform switches.	10.3(1)F	Guidelines and Limitations for Online Diagnostics , on page 288 Runtime or Health Monitoring Diagnostics , on page 282

Feature	Description	Changed in Release	Where Documented
Command to disable L2FM MAC learning behavior during loops	MAC move policy is supported on Cisco Nexus 9300-X Cloud Scale Switches.	10.3(1)F	Configuring MAC Move Policy, on page 337
ePBR L2 - GX	The provider VLAN tagging is supported on the Cisco Nexus 9300-GX, N9K-C9504-FM-G, and N9K-C9508-FM-G switches and N9K-X9716D-GX line cards.	10.3(1)F	Guidelines and Limitations for TAP Aggregation, on page 468 Configuring Selective Q-in-Q with Provider VLANs, on page 475



CHAPTER 2

Overview

This chapter describes the system management features that you can use to monitor and manage Cisco NX-OS devices.

- [Licensing Requirements, on page 7](#)
- [Supported Platforms, on page 7](#)
- [Cisco NX-OS Device Configuration Methods, on page 8](#)
- [Network Time Protocol, on page 9](#)
- [Cisco Discovery Protocol, on page 9](#)
- [Session Manager, on page 9](#)
- [Scheduler, on page 9](#)
- [SNMP, on page 10](#)
- [Online Diagnostics, on page 10](#)
- [Onboard Failure Logging, on page 10](#)
- [SPAN, on page 10](#)
- [ERSPAN, on page 10](#)
- [LLDP, on page 10](#)
- [MPLS Stripping, on page 11](#)
- [sFlow, on page 11](#)
- [SMUs, on page 11](#)
- [Virtual Device Contexts, on page 11](#)
- [Troubleshooting Features, on page 11](#)

Licensing Requirements

For a complete explanation of Cisco NX-OS licensing recommendations and how to obtain and apply licenses, see the [Cisco NX-OS Licensing Guide](#) and the [Cisco NX-OS Licensing Options Guide](#).

Supported Platforms

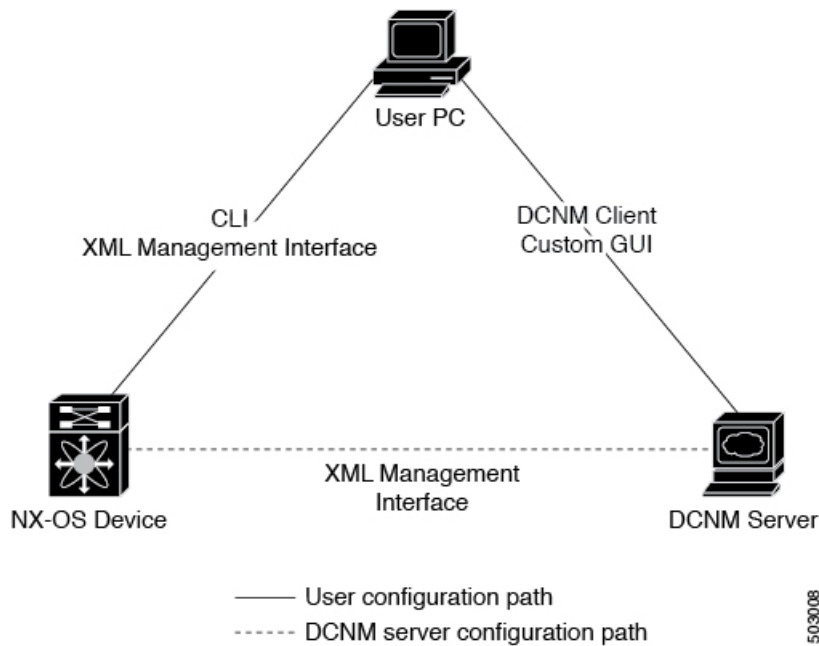
Use the [Nexus Switch Platform Support Matrix](#) to know from which Cisco NX-OS releases various Cisco Nexus 9000 and 3000 switches support a selected feature.

Cisco NX-OS Device Configuration Methods

You can configure devices using direct network configuration methods or web services hosted on a Cisco Data Center Network Management (DCNM) server.

This figure shows the device configuration methods available to a network user.

Figure 1: Cisco NX-OS Device Configuration Methods



This table lists the configuration method and the document where you can find more information.

Table 2: Configuration Methods Book Links

Configuration Method	Document
CLI from a Secure Shell (SSH) session, a Telnet session, or the console port	
Cisco DCNM client	<i>Cisco DCNM Fundamentals Guide</i>

Configuring with CLI or XML Management Interface

You can configure Cisco NX-OS devices using the command-line interface (CLI) or the XML management interface over Secure Shell (SSH) as follows:

- CLI from an SSH session, a Telnet session, or the console port—You can configure devices using the CLI from an SSH session, a Telnet session, or the console port. SSH provides a secure connection to the

device. For more information, see the *Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*.

- XML management interface over SSH—You can configure devices using the XML management interface, which is a programmatic method based on the NETCONF protocol that complements the CLI functionality. For more information, see the *Cisco NX-OS XML Management Interface User Guide*.

Configuring with Cisco DCNM

You can configure Cisco NX-OS devices using the Cisco DCNM client, which runs on your local PC and uses web services on the Cisco DCNM server. The Cisco DCNM server configures the device over the XML management interface. For more information about the Cisco DCNM client, see the [Cisco DCNM Fundamentals Guide](#).

Network Time Protocol

The Network Time Protocol (NTP) synchronizes the time of day among a set of distributed time servers and clients so that you can correlate time-specific information, such as system logs, received from the devices in your network.

Cisco Discovery Protocol

You can use the Cisco Discovery Protocol (CDP) to discover and view information about all Cisco equipment that is directly attached to your device. CDP runs on all Cisco-manufactured equipment including routers, bridges, access and communication servers, and switches. CDP is media and protocol independent, and gathers the protocol addresses of neighboring devices, discovering the platform of those devices. CDP runs over the data link layer only. Two systems that support different Layer 3 protocols can learn about each other.

Session Manager

Session Manager allows you to create a configuration and apply it in batch mode after the configuration is reviewed and verified for accuracy and completeness.

Scheduler

The scheduler allows you to create and manage jobs such as routinely backing up data or making quality of service (QoS) policy changes. The scheduler can start a job according to your needs—only once at a specified time or at periodic intervals.

SNMP

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

Online Diagnostics

Cisco Generic Online Diagnostics (GOLD) define a common framework for diagnostic operations across Cisco platforms. The online diagnostic framework specifies the platform-independent fault-detection architecture for centralized and distributed systems, including the common diagnostics CLI and the platform-independent fault-detection procedures for boot-up and run-time diagnostics. The platform-specific diagnostics provide hardware-specific fault-detection tests and allow you to take appropriate corrective action in response to diagnostic test results.

Onboard Failure Logging

You can configure a device to log failure data to persistent storage, which you can retrieve and display for analysis at a later time. This on-board failure logging (OBFL) feature stores failure and environmental information in nonvolatile memory on the module. This information is useful for analysis of failed modules.

SPAN

You can configure an Ethernet Switched Port Analyzer (SPAN) to monitor traffic in and out of your device. The SPAN features allow you to duplicate packets from source ports to destination ports.

ERSPAN

Encapsulated Remote Switched Port Analyzer (ERSPAN) is used to transport mirrored traffic in an IP network. ERSPAN supports source ports, source VLANs, and destinations on different switches, which provide remote monitoring of multiple switches across your network.

To configure an ERSPAN source session, you associate a set of source ports or VLANs with a destination IP address, ERSPAN ID number, and virtual routing and forwarding (VRF) name.

LLDP

Link Layer Discovery Protocol (LLDP) is a vendor-neutral, one-way device discovery protocol that allows 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. You can enable LLDP globally or per interface.

MPLS Stripping

MPLS stripping provides the ability to strip MPLS labels from packets, enabling non-MPLS-capable network monitoring tools to monitor packets.

sFlow

Sampled flow (sFlow) allows you to monitor real-time traffic in data networks that contain switches and routers and to forward the sample data to a central data collector.

SMUs

A software maintenance upgrade (SMU) is a package file that contains fixes for a specific defect. SMUs are created to respond to immediate issues and do not include new features. SMUs are not an alternative to maintenance releases. They provide a quick resolution of immediate issues. All defects fixed by SMUs are integrated into the maintenance releases.

Virtual Device Contexts

Cisco NX-OS can segment operating system and hardware resources into virtual device contexts (VDCs) that emulate virtual devices. The Cisco Nexus 9000 Series switches currently do not support multiple VDCs. All switch resources are managed in the default VDC.

Troubleshooting Features

Cisco NX-OS provides troubleshooting tools such as ping, traceroute, Ethanalyzer, and the Blue Beacon feature.

When a service fails, the system generates information that can be used to determine the cause of the failure. The following sources of information are available:

- Every service restart generates a syslog message of level LOG_ERR.
- If the Smart Call Home service is enabled, every service restart generates a Smart Call Home event.
- If SNMP traps are enabled, the SNMP agent sends a trap when a service is restarted.
- When a service failure occurs on a local module, you can view a log of the event by entering the **show processes log** command in that module. The process logs are persistent across supervisor switchovers and resets.
- When a service fails, a system core image file is generated. You can view recent core images by entering the **show cores** command on the active supervisor. Core files are not persistent across supervisor switchovers and resets, but you can configure the system to export core files to an external server using the file transfer utility Trivial File Transfer Protocol (TFTP) by entering the **system cores** command.
- CISCO-SYSTEM-MIB contains a table for cores (cseSwCoresTable).



CHAPTER 3

Two-stage Configuration Commit

This chapter describes how to enable two-stage configuration commit mode on the Cisco NX-OS device.

This chapter includes the following sections:

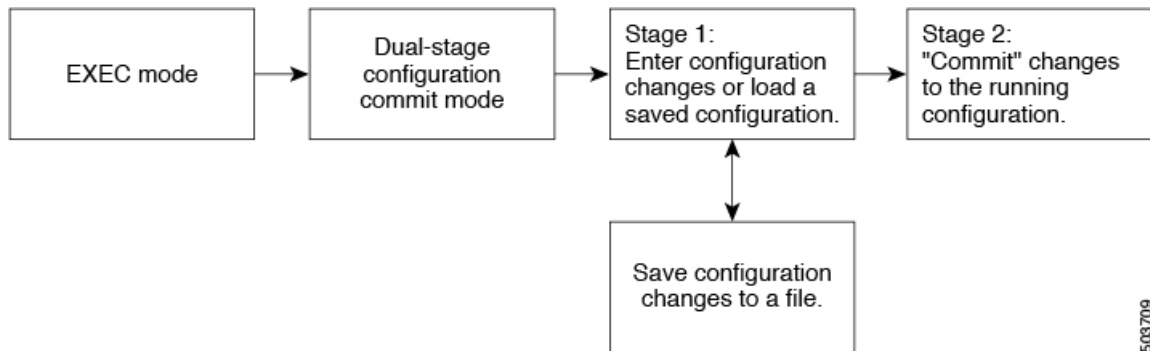
- [About Two-stage Configuration Commit, on page 13](#)
- [Guidelines and Limitations, on page 14](#)
- [Configuring in Two-Stage Configuration Commit Mode, on page 15](#)
- [Aborting the Two-Stage Configuration Commit Mode, on page 21](#)
- [Displaying Commit IDs, on page 21](#)
- [Rollback Capability, on page 22](#)
- [Viewing Current Session Configurations, on page 22](#)

About Two-stage Configuration Commit

In an interactive session, when you run a command, it's executed and it changes the running configuration. This behaviour is known as one-stage configuration commit. In the confirm-commit or the two-stage configuration commit, changes in configurations are stored in a staging database. These changes don't affect the running configuration until you run the **commit** command. This two-stage process creates a target configuration session, where you can make, edit, and verify configuration changes before committing them to the running state of the switch. You can also commit the changes for a time period you specify before you commit them permanently. After the specified time period, the switch reverts to the previous configuration if you don't run the **commit** command. When a commit is successful, you can view the commit information that includes the commit ID, username, and timestamp.

The following figure shows the two-stage configuration commit process.

Figure 2: Two-Stage Configuration Commit Process



503709

Guidelines and Limitations

Two-stage configuration commit has the following configuration guidelines and limitations:

- This feature is supported only for a CLI interface in a user-interactive session.
- Before you run any feature-related configuration commands, enable the feature using the **feature** command and commit it using the **commit** command.
- Two-stage configuration commit mode doesn't support other modes like maintenance mode, scheduler mode, or virtual mode.
- When you're in the two-stage configuration commit mode, avoid editing configurations in one-stage configuration commit mode from different sessions at the same time.
- Review the configurations using the **show configuration** command before committing the changes.
- Show configuration displays the staged configs:
 - It displays the real difference, that is yes and no form of the same command will result in empty config.
 - It is recommended to issue the exact no form of the cli to negate the config.
Example: to negate 'ip address x' config, user has to give 'no ip address x' instead of 'no ip address'.
- Interface layer change commands (switchport/no switchport) should be issued explicitly.
- Any invalid config in the session should manually be removed by the user before attempting commit. If could not remove manually clear the session and start a new session.
- If the verification fails, edit and retry the commit.
- If the commit fails, the configuration rolls back to the previous configuration.
- Configurations that you don't commit aren't saved after you reload the switch.
- This feature doesn't support commits with NX-API, EEM, PPM and Netconf.
- You can have only one active two-stage configuration commit session at a given time.

Configuring in Two-Stage Configuration Commit Mode

To enable a feature in the two-stage configuration commit mode, perform the following steps:



Note In this procedure, the BGP feature is enabled as an example.

Procedure

	Command or Action	Purpose
Step 1	configure dual-stage Example: <pre>switch# configure dual-stage switch(config-dual-stage) #</pre>	Creates a new target configuration session. Note The target configuration isn't a copy of the running configuration. It has only the configuration commands entered during the target configuration session.
Step 2	feature <i>feature_name</i> Example: <pre>switch(config-dual-stage) # feature bgp switch(config-dual-stage) #</pre>	Enables the feature. Note - You can enable the feature even before entering the two-stage configuration commit mode. - You can't combine feature-related commands in a commit if the feature isn't already enabled.
Step 3	commit [confirmed <i>seconds</i>] Example: <pre>switch(config-dual-stage-router) # commit confirmed 30 Verification Succeeded. Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time. Configuration committed by user 'admin' using Commit ID : 1000000001 switch(config-dual-stage) # switch(config-dual-stage) # commit Confirming commit for trial session. switch(config-dual-stage) #</pre> Example:	Commits changes to the running configuration. confirmed: Commits the changes to the running configuration. seconds: Commits the configuration in global configuration mode on a trial basis for a minimum of 30 seconds and a maximum of 65535 seconds. Note If you enter a trial period, run the commit command to confirm the configuration. If you don't run the commit command, the switch reverts to the previous configuration after the trial period.

	Command or Action	Purpose
	<pre>switch(config-dual-stage)# hostname example-switch switch(config-dual-stage)# commit Verification Succeeded.</pre> Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time. Configuration committed by user 'admin' using Commit ID : 1000000002 <pre>example-switch(config-dual-stage)#</pre>	
Step 4	Example: <pre>switch(config-dual-stage)# router bgp 64515.46 switch(config-dual-stage-router)# switch(config-dual-stage-router)# router-id 141.8.139.131 switch(config-dual-stage-router)#</pre>	Run any feature-related commands that are supported in this configuration mode.
Step 5	show configuration Example: <pre>switch(config-dual-stage-router)# show configuration ! Cached configuration ! router bgp 64515.46 router-id 141.8.139.131</pre>	Displays the target configuration. Note You can run this command only in the dual-stage configuration mode.
Step 6	commit [confirmed seconds] Example: <pre>switch(config-dual-stage-router)# commit</pre> Verification Succeeded. Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time. Configuration committed by user 'admin' using Commit ID : 1000000003	Commits changes to the running configuration.
Step 7	(Optional) show configuration commit [changes] commit-id Example: <pre>switch(config-dual-stage-router)# show configuration commit changes 1000000003 *** /bootflash/.dual-stage/1000000003.tmp Fri Mar 19 10:59:00 2021 --- /bootflash/.dual-stage/1000000003 Fri Mar 19 10:59:05 2021 ***** *** 378,383 **** --- 378,385 ----</pre>	Displays commit-related information. Only the last 50 commits or the commit files stored in the reserved disk space are saved. The reserved disk space is 20 MB. All the commit sessions will be removed when you reload the switch. However, the commit IDs are not removed. Also, these commit IDs will not be removed in case of write, erase, and reload. Use the show configuration commit changes commit-id command to view only the changes

	Command or Action	Purpose
	<pre> line console line vty boot nxos bootflash:/nxos64.10.1.1.44.bin + router bgp 64515.46 + router-id 141.8.139.131 xml server timeout 1200 no priority-flow-control override-interface mode off Example: switch(config-dual-stage)# show configuration commit 1000000003 feature bgp router bgp 64515.46 router-id 141.8.139.131 . . . </pre>	in the current session of the commit you specify. Use the show configuration commit commit-id command to view the complete configurations in the commit that you specify, along with few class-map policies. These class-map policies are not new policies but hidden policies. To view the hidden policies, use the show run all command.
Step 8	(Optional) save configuration filename Example: <pre> switch(config-dual-stage)# save configuration bootflash:test.cfg </pre>	Saves the target configurations to a separate file without committing them to the running configuration. Note • You can load the target configuration files later, modify, or commit. The file will be saved in bootflash. • You can view the configuration file you saved by running the show configuration file filename command. • Some of the user-specific information will be masked based on the user role. • Configs saved in dual stage mode is an encrypted file and can be viewed only using #show configuration file <> and not using #show file <>.
Step 9	(Optional) load filename Example: <pre> switch (config-dual-stage)# show configuration ! Cached configuration switch (config-dual-stage)# load test.cfg switch (config-dual-stage-router)# show configuration ! Cached configuration ! router bgp 1 switch(config-dual-stage-router)# </pre>	Loads a target configuration that you saved. After loading a file, you can modify it or commit it to the running configuration. To save the changes, use the save configuration filename command. You can load a target configuration that you saved using only the save configuration filename command.

	Command or Action	Purpose
Step 10	(Optional) clear configuration Example: <pre>switch(config-dual-stage)# show configuration ! Cached configuration ! router bgp 64515.46 router-id 141.8.139.131 switch (config-dual-stage)# clear configuration switch (config-dual-stage)# show configuration ! Cached configuration switch (config-dual-stage)#</pre>	Clears changes made to the target configuration without terminating the configuration session. It deletes any configuration changes that aren't committed.
Step 11	end Example: <pre>switch(config-dual-stage-if)# end Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]</pre>	Exits the global dual stage configuration mode. If you end a configuration session without committing the configuration changes, you'll be prompted to save changes, discard changes, or cancel the action: • Yes: Commits the configuration changes and exit configuration mode • No: Exits the configuration mode without committing the configuration changes • Cancel: Remains in configuration mode without committing the configuration changes Note • If you choose to exit when a confirm commit timer is running, the same options are displayed. If you still chose to exit, the trial configuration rolls back instantly. • If the default session times out before the timer expires, the trial configuration rolls back before exiting the session. In this case, no warning message appears.
Step 12	show configuration dual-stage sessions Example: <pre>switch(config-dual-stage)# show configuration dual-stage sessions SNo. Session Line User Date ----- 1 8671-17101913 /dev/ttyS0 admin</pre>	Before you start a configuration session, you must check if there are other configuration sessions in progress. Only single user is allowed to enter the dual stage configuration mode. Therefore, you need to exit the previous session before starting a new one. There are as many as 32 interactive VSH sessions possible, and the show command displays the

	Command or Action	Purpose
	<pre> Wed Feb 17 10:56:00 2021 switch(config-dual-stage)# end switch# show configuration dual-stage sessions There are no active dual stage sessions switch# </pre>	PID and line information of the dual stage session. Note Dual stage mode will be accessible only after System ready.
Step 13	clear configuration commits diskspace Example: <pre> Southlake-2# clear configuration commits diskspace ? <1-20971> Number of Kilo Bytes of disk space to free Southlake-2# clear configuration commits diskspace 100 Deleting 7 rollback points from '1000005557' to '1000005563' 101 KB of disk space will be freed. Continue with deletion (yes/no)? [no] y Southlake-2# </pre>	You can delete the oldest configuration commitIDs by entering the clear configuration commits command. The clear configuration commits command must be followed by either the amount of disk space to reclaim or the number of commitIDs to delete. To reclaim disk space from the oldest commitIDs, enter the clear configuration commits command followed by the diskpace keyword and number of kilobytes to reclaim.
Step 14	clear configuration commits oldest Example: <pre> switch(config-dual-stage)# clear configuration commits oldest 10 Deleting 10 rollback points '1000000030' to '1000000039' 125 KB of disk space will be freed. Continue with deletion (yes/no)? [no] n </pre>	To delete a specific number of the oldest commitIDs, enter the clear configuration commits command followed by the oldest keyword and number of commitIDs to delete.
Step 15	Show configuration failed Example: <pre> switch(config-dual-stage-if)# commit Verification Succeeded. Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time. Failed to commit one or more configuration items. Commit Failed, Rolling back ... switch(config-dual-stage)# switch(config-dual-stage)# show configuration failed `config terminal` `router bgp 100 ` `neighbor 2.2.2.2 ` `bfd ` Syntax error while parsing 'bfd ' </pre>	Configuration changes are semantically verified during the commit operation, and begins the actual backend commit once the verification is succeeded. A message appears if one or more configuration entry fails while committing. To display an error message and description for a failed configuration, enter the show configuration failed command. This will display the configuration block that failed in last commit. Configuration block is to preserve the configuration context.

	Command or Action	Purpose
	<pre> `neighbor 3.3.3.3 ` `bfd ` Syntax error while parsing 'bfd ' `interface port-channel23 ` `bfd ` Syntax error while parsing 'bfd ' `end` `end` switch(config-dual-stage)# </pre>	
Step 16	show configuration failed noerrors Example: <pre> switch(config-dual-stage)# show configuration failed noerror router bgp 100 neighbor 2.2.2.2 bfd neighbor 3.3.3.3 bfd interface port-channel23 bfd switch(config-dual-stage)# </pre>	To display only the errored config (without a description) for a failed configuration block, enter the show configuration failed noerrors command.
Step 17	load configuration failed commit Example: <pre> switch(config-dual-stage)# load configuration failed commit switch(config-dual-stage-if)# sh configuration ! Cached configuration ! router bgp 100 neighbor 2.2.2.2 bfd ! interface port-channel23 bfd switch(config-dual-stage-if)# </pre>	If the router displays a verification failure message during commit, the configuration changes are not lost. You can modify the target configuration and commit again. But, if the router displays a configuration failure message (backend error) when you attempt to commit a configuration change, the configuration session will reset. But, while you remain in dual-stage configuration mode, you can reload the failed configuration block into the target configuration, correct the errors, and commit the changes. To load a failed configuration, enter the load configuration failed commit command. After recovery, correct and commit the configuration or save it to a file to avoid losing it. Please note that while loading, syntactically wrong configurations will get ignored. You can use 'show configuration' to view the target configuration.

Aborting the Two-Stage Configuration Commit Mode

When you abort a configuration session, uncommitted changes are discarded and the configuration session ends. No warning appears before the configuration changes are deleted.

```
switch(config-dual-stage)# router bgp 1
switch(config-dual-stage-router)# neighbor 1.2.3.4
switch(config-dual-stage-router-neighbor)# remote-as 1
switch(config-dual-stage-router-neighbor)# show configuration
! Cached configuration
!
router bgp 1
neighbor 1.2.3.4
remote-as 1
switch(config-dual-stage-router-neighbor)# show run bgp

!Command: show running-config bgp
!Running configuration last done at: Wed Mar 17 16:17:40 2021
!Time: Wed Mar 17 16:17:55 2021

version 10.1(2) Bios:version
feature bgp

switch(config-dual-stage-router-neighbor)# abort
switch# show run bgp

!Command: show running-config bgp
!Running configuration last done at: Wed Mar 17 16:18:00 2021
!Time: Wed Mar 17 16:18:04 2021

version 10.1(2) Bios:version
feature bgp

switch#
```

Displaying Commit IDs

At each successful commit, the commit ID is displayed in the syslog. The total number of commit IDs saved in the system depends on the configuration size and the disk space available. However, the maximum number of commit IDs stored at any given time is 50.

Use the **show configuration commit list** command to view information about the last 50 commit IDs. Each entry shows the user who committed configuration changes, the connection used to execute the commit, and commit ID timestamp.

```
switch# show configuration commit list
```

SNo.	Label/ID	User	Line	Client	Time Stamp
1	1000000001	admin	/dev/ttyS0	CLI	Wed Jul 15 15:21:37 2020
2	1000000002	admin	/dev/ttyS0	Rollback	Wed Jul 15 15:22:15 2020
3	1000000003	admin	/dev/pts/0	CLI	Wed Jul 15 15:23:08 2020
4	1000000004	admin	/dev/pts/0	Rollback	Wed Jul 15 15:23:46 2020

Rollback Capability

You can rollback the configuration to any of the previous successful commits. Use the **rollback configuration** command to rollback to any of the last 50 commits.

```
switch# rollback configuration to ?
1000000015
1000000016
1000000017
```

```
:
```

```
switch#
```

Each commit ID acts as a (checkpoint or) rollback point. You can rollback to any given commit ID. When you roll back the configuration to a specific rollback point, you undo all configuration changes made during the session identified by the commitID for that rollback point, and you undo all configuration changes made after that point. The rollback process rolls back the configuration and commits the rolled-back configuration. The rollback process also creates a new rollback point (commit ID) so that you can roll back the configuration to the previous configuration.

```
switch(config-dual-stage)# rollback configuration to 1000000002
Rolling back to commitID :1000000002
ADVISORY: Rollback operation started...
Modifying running configuration from another VSH terminal in parallel
is not recommended, as this may lead to Rollback failure.

Configuration committed by rollback using Commit ID : 1000000004
switch(config-dual-stage)#
```

Viewing Current Session Configurations

You can view the current session configuration using the **show configuration** command. This command is supported only in the dual-stage mode. The session configuration is cleared if a commit fails.

```
switch(config-dual-stage-cmap)# show configuration
! Cached configuration
!
class-map type control-plane match-any copp-s-ipmcmis
class-map type control-plane match-any copp-s-l2switched
class-map type control-plane match-any copp-s-l3destmiss
switch(config-dual-stage-cmap)#
```

If there is no configuration, the following message appears:

```
switch(config-dual-stage)# show configuration
! Cached configuration
switch(config-dual-stage)# commit
No configuration changes to commit.
switch(config-dual-stage)#
```



CHAPTER 4

Configuring Switch Profiles

This chapter describes how to configure switch profiles on the Cisco Nexus 9000 Series switches.

- [About Switch Profiles, on page 23](#)
- [Guidelines and Limitations for Switch Profiles, on page 25](#)
- [Configuring Switch Profiles, on page 27](#)
- [Adding or Modifying Switch Profile Commands , on page 29](#)
- [Importing a Switch Profile, on page 30](#)
- [Importing Configurations in a vPC Topology, on page 32](#)
- [Isolating a Peer Switch, on page 33](#)
- [Deleting a Switch Profile, on page 33](#)
- [Manually Correcting Mutex and Merge Failures, on page 34](#)
- [Verifying the Switch Profile Configuration, on page 34](#)
- [Configuration Examples for Switch Profiles, on page 35](#)

About Switch Profiles

Several applications require consistent configuration across devices in the network. For example, with a virtual port channel (vPC), you must have identical configurations. Mismatched configurations can cause errors or misconfigurations that can result in service disruptions. The configuration synchronization (config-sync) feature allows you to configure one switch profile and have the configuration be automatically synchronized to the peer switch.

A switch profile provides the following benefits:

- Allows configurations to be synchronized between switches.
- Merges configurations when connectivity is established between two switches.
- Provides control of exactly which configuration gets synchronized.
- Ensures configuration consistency across peers through merge and mutual-exclusion checks.
- Provides verify and commit semantics.
- Allows for migrating existing vPC configurations to a switch profile.

Switch Profile Configuration Modes

The switch profile feature includes the following configuration modes:

- Configuration synchronization mode (config-sync)
- Switch profile mode (config-sync-sp)
- Switch profile import mode (config-sync-sp-import)

Configuration Synchronization Mode

The configuration synchronization mode (config-sync) allows you to create switch profiles.

Switch Profile Mode

The switch profile mode (config-sync-sp) allows you to add supported configuration commands to a switch profile temporary buffer that is later synchronized with a peer switch. Commands that you enter in the switch profile mode are not executed until you enter the **commit** command. Although the syntax of the commands are validated when you enter them, there is no guarantee that the commands will be successful when you enter the **commit** command.

Switch Profile Import Mode

The switch profile import mode (config-sync-sp-import) allows you to import existing switch configurations from the running configuration to a switch profile and specify which commands you want to include in that profile. This option is especially useful when you upgrade from a Cisco NX-OS release that does not support switch profiles to a release that does.

Cisco recommends that you import the necessary configurations from the running configuration using the switch profile import mode and commit the changes before making any additional changes in the switch profile or global configuration mode. Otherwise, you might jeopardize the import, requiring you to abandon the current import session and perform the process again. For more information, see [Importing a Switch Profile](#), on page 30.

Configuration Validation

Two types of configuration validation checks can identify switch profile failures:

- Mutual exclusion checks
- Merge checks

Mutual Exclusion Checks

The mutual exclusion of configuration commands is enforced in order to avoid duplicate commands in the config-sync and global configuration modes. When you commit the configuration of a switch profile, mutual exclusion (mutex) checks are performed on the local switch as well as the peer switch (if configured). If no failures are reported on both switches, the commit is accepted and pushed into the running configuration.

A command that is included in a switch profile cannot be configured outside of the switch profile.

If a mutex check identifies errors, they are reported as mutex failures, and they must be manually corrected. For details, see [Manually Correcting Mutex and Merge Failures](#), on page 34.

The following exceptions apply to the mutual exclusion policy:

- Interface configuration—An interface configuration can be partially present in a switch profile and partially present in the running configuration as long as there are no conflicts.
- Shutdown/no shutdown
- System QoS

Merge Checks

Merge checks are done on the peer switch that is receiving a configuration. The merge checks ensure that the received configuration does not conflict with the switch profile configuration that already exists on the receiving switch. The merge check occurs during the verify or commit process. Errors are reported as merge failures and must be manually corrected. For details, see [Manually Correcting Mutex and Merge Failures, on page 34](#).

When one or both switches are reloaded and the configurations are synchronized for the first time, the merge check verifies that the switch profile configurations are identical on both switches. Differences in the switch profiles are reported as merge errors and must be manually corrected.

Software Upgrades and Downgrades with Switch Profiles

You must delete the switch profile when downgrading from a Cisco NX-OS release that supports switch profiles to a release that does not.

When you upgrade from an earlier release to a Cisco NX-OS release that supports switch profiles, you have the option to move some of the running-configuration commands to a switch profile. For details, see [Switch Profile Import Mode, on page 24](#).

An upgrade can occur if there are buffered (uncommitted) configurations; however, the uncommitted configurations will be lost.

Guidelines and Limitations for Switch Profiles

Switch profiles have the following configuration guidelines and limitations:

- Beginning with Cisco NX-OS Release 9.3(3), the **mtu** command is supported in the interface configuration mode through the switch-profiles configuration mode.
- Switch profiles are supported only on Cisco Nexus 9300 Series switches. Cisco Nexus 9500 Series switches do not support switch profiles.
- You can only enable configuration synchronization using the mgmt0 interface.
- When using config-sync in a virtual peer-link environment, note the following limitations:
 - To initiate a config-sync session with a virtual peer link, be sure to configure a loopback IP address instead of a management IP address between the peer switches.
 - You cannot perform a configuration synchronization between a multichassis EtherChannel trunk (MCT) configuration and a virtual peer-link configuration. This config-sync operation is not supported.

- You must configure synchronized peers with the same switch profile name.
- Commands that are qualified for a switch profile configuration are allowed to be configured in the configuration switch profile mode (config-sync-sp).
- Supported switch profile commands relate to vPC commands.
- Only one switch profile session can be in progress at a time. Attempts to start another session will fail.
- Command changes made from the global configuration mode are blocked when a switch profile session is in progress.
- When you enter the **commit** command and a peer switch is reachable, the configuration is applied to both peer switches or neither switch. If a commit failure occurs, the commands remain in the switch profile buffer. You can then make necessary corrections and try the commit again.
- The configuration synchronization (**config-sync**) mode is an L2 mode parallel to the config-terminal mode (**config t**). Config-sync uses the switch-profile to update **config t** mode in the same switch as well as the peer switch. To prevent sync issues in **switch-profile** mode, Cisco recommends that you perform a commit action after each CLI command before overriding, or replacing the current CLI command.

For example, if you want to overwrite **CLI_command_A** and change it to **CLI_command_B**, commit **CLI_command_A** first, then configure **CLI_command_B** and perform another commit action.

```
switch# conf sync
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-sync)# switch-profile test
Resyncing db before starting Switch-profile.Re-synchronization of switch-profile db
takes a few minutes...
Re-synchronize switch-profile db completed successfully.
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)#
switch(config-sync-sp)# int e 1/3
switch(config-sync-sp-if)# switchport trunk allowed vlan 100-150
switch(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch(config-sync)#
switch(config-sync)# switch-profile test
Resyncing db before starting Switch-profile.Re-synchronization of switch-profile db
takes a few minutes...
Re-synchronize switch-profile db completed successfully.
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)#
switch(config-sync-sp)# int e 1/3
switch(config-sync-sp-if)# switchport trunk allowed vlan 45-90
switch(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch(config-sync)# end
switch#
```

- Layer 3 commands are not supported.

The config-sync feature has the following guidelines and limitations:

- Port-channels created in the switch profile mode should not be configured using global configuration (config terminal) mode.
- If a port-channel is created in global configuration mode, channel groups including member interfaces must also be created using global configuration mode.
- Port-channels that are configured within the switch profile mode may have members both inside and outside of a switch profile.
- If you want to import a member interface to a switch profile, the port-channel that corresponds with the member interface must also be present within the switch profile.
- For "no system default switchport" configuration at global level, the "switchport" command under port-channel is also considered for mutual exclusion.

Configuring Switch Profiles

You can create and configure a switch profile on the local switch and then add a second switch that will be included in the synchronization.

You must create the switch profile with the same name on each switch, and the switches must configure each other as a peer. When connectivity is established between switches with the same active switch profile, the switch profiles are synchronized.

Procedure

Step 1 **configure terminal****Example:**

```
switch# configure terminal
switch(config)#
```

Enters global configuration mode.

Step 2 Required: **cfs ipv4 distribute****Example:**

```
switch(config)# cfs ipv4 distribute
```

Enables Cisco Fabric Services (CFS) distribution between the peer switches.

Step 3 Required: **config sync****Example:**

```
switch(config)# config sync
switch(config-sync)#
```

Enters the configuration synchronization mode.

Step 4 Required: **switch-profile name****Example:**

```
switch(config-sync)# switch-profile abc
switch(config-sync-sp)#
```

Configures the switch profile, names the switch profile, and enters the switch profile configuration mode.

Step 5 Required: **[no] sync-peers destination ip-address**

Example:

```
switch(config-sync-sp)# sync-peers destination 10.1.1.1
```

Adds a switch to the switch profile. The destination IP address is the IP address of the switch that you want to synchronize.

The **no** form of this command removes the specified switch from the switch profile.

Note

You need to wait for peer switches to show the switch-profile status of "In sync" before any commit is done.

Step 6 Required: For Cisco Nexus 3164Q switches only, follow these steps:

a) **interface type slot/port**

Example:

```
switch(config-sync-sp)# interface ethernet 1/1  
switch(config-sync-sp-if)#
```

Enters the switch profile interface configuration mode.

b) **switchport**

Example:

```
switch(config-sync-sp-if)# switchport
```

Changes a Layer 3 interface into a Layer 2 interface.

c) **exit**

Example:

```
switch(config-sync-sp-if)# exit  
switch(config-sync-sp)#
```

Exits the switch profile interface configuration mode.

d) **commit**

Example:

```
switch(config-sync-sp)# commit
```

Commits the current configuration.

Note

Verify that the switch-profile status shows as "In sync" before any commit is done.

Step 7 (Optional) **end**

Example:

```
switch(config-sync-sp)# end  
switch#
```

Exits the switch profile configuration mode and returns to EXEC mode.

Step 8 (Optional) **show switch-profile name status**

Example:

```
switch# show switch-profile abc status
```

Displays the switch profile on the local switch and the peer switch information.

Step 9 (Optional) **show switch-profile** *name* **peer** *ip-address*

Example:

```
switch# show switch-profile abc peer 10.1.1.1
```

Displays the switch profile peer configuration.

Step 10 (Optional) **copy running-config startup-config**

Example:

```
switch# copy running-config startup-config
```

Copies the running configuration to the startup configuration.

Adding or Modifying Switch Profile Commands

After you configure a switch profile on the local and the peer switch, you must add and commit the supported commands to the switch profile.

Commands that are added or modified are buffered until you enter the **commit** command. Commands are executed in the same order in which they are buffered. If there is an order dependency for certain commands (for example, a QoS policy must be defined before being applied), you must maintain that order; otherwise, the commit might fail. You can use utility commands, such as the **show switch-profile** *name* **buffer** command, the **buffer-delete** command, and the **buffer-move** command, to change the buffer and correct the order of already entered commands.

Procedure

	Command or Action	Purpose
Step 1	Required: config sync Example: switch# config sync switch(config-sync)#	Enters the configuration synchronization mode.
Step 2	Required: switch-profile <i>name</i> Example: switch(config-sync)# switch-profile abc switch(config-sync-sp)#	Configures the switch profile, names the switch profile, and enters the switch profile configuration mode.
Step 3	Required: <i>command</i> Example: switch(config-sync-sp)# interface Port-channel100 switch(config-sync-sp-if)# speed 1000 switch(config-sync-sp-if)# interface Ethernet1/1 switch(config-sync-sp-if)# speed 1000	Adds a command to the switch profile.

	Command or Action	Purpose
	<pre>switch(config-sync-sp-if) # channel-group 100 switch(config-sync-sp-if) # exit switch(config-sync-sp) #</pre>	
Step 4	(Optional) show switch-profile name buffer Example: <pre>switch(config-sync-sp) # show switch-profile abc buffer</pre>	Displays the configuration commands in the switch profile buffer.
Step 5	Required: verify Example: <pre>switch(config-sync-sp) # verify</pre>	Verifies the commands in the switch profile buffer.
Step 6	Required: commit Example: <pre>switch(config-sync-sp) # commit</pre>	Saves the commands in the switch profile and synchronizes the configuration with the peer switch. This command also does the following: • Triggers the mutex check and the merge check to verify the synchronization. • Creates a checkpoint with a rollback infrastructure. • Executes a rollback on all switches if an application failure occurs on any of the switches in the switch profile. • Deletes the checkpoint.
Step 7	(Optional) end Example: <pre>switch(config-sync-sp) # end switch#</pre>	Exits the switch profile configuration mode and returns to EXEC mode.
Step 8	(Optional) show switch-profile name status Example: <pre>switch# show switch-profile abc status</pre>	Displays the status of the switch profile on the local switch and the status on the peer switch.
Step 9	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Importing a Switch Profile

You can import a switch profile based on the set of commands that you want to import.

Before you begin

Make sure that the switch profile buffer is empty before you import commands to a switch profile.

Procedure

	Command or Action	Purpose
Step 1	(Optional) Configure the interface that will be imported in Step 4. Example: <pre>switch(config)# interface ethernet 1/2 switch(config-if)# switchport switch(config-if)# switchport mode trunk switch(config-if)# switchport trunk allowed vlan 12 switch(config-if)# speed 10000 switch(config-if)# spanning-tree port type edge trunk switch(config)# end switch#</pre>	Enters configuration synchronization mode.
Step 2	config sync Example: <pre>switch# config sync switch(config-sync)#</pre>	Enters configuration synchronization mode.
Step 3	Required: switch-profile name Example: <pre>switch(config-sync)# switch-profile abc switch(config-sync-sp)#</pre>	Configures the switch profile, names the switch profile, and enters the switch profile configuration mode.
Step 4	Required: import [interface interface port/slot running-config] Example: <pre>switch(config-sync-sp)# import interface ethernet 1/2 switch(config-sync-sp-import)#</pre>	Identifies the commands that you want to import and enters the switch profile import mode. The following options are available: • Entering the import command without any options adds the selected commands to the switch profile. • The import interface option adds the supported commands for a specified interface. • The running-config option adds supported system-level commands. Note If new commands are added during the import, the switch profile remains unsaved, and the switch remains in the switch profile import mode.

	Command or Action	Purpose
Step 5	Required: commit Example: <code>switch(config-sync-sp-import)# commit</code>	Imports the commands and saves the commands to the switch profile.
Step 6	(Optional) abort Example: <code>switch(config-sync-sp-import)# abort</code>	Aborts the import process.
Step 7	(Optional) end Example: <code>switch(config-sync-sp-import)# end</code> <code>switch#</code>	Exits the switch profile import mode and returns to EXEC mode.
Step 8	(Optional) show switch-profile Example: <code>switch# show switch-profile</code>	Displays the switch profile configuration.
Step 9	(Optional) copy running-config startup-config Example: <code>switch# copy running-config</code> <code>startup-config</code>	Copies the running configuration to the startup configuration.

Importing Configurations in a vPC Topology

You can import configurations in a two-switch vPC topology.



Note For specific information on the following steps, see the appropriate sections in this chapter.

1. Configure the switch profile with the same name on both switches.
2. Import the configurations to both switches independently.



Note Make sure that the configuration moved to the switch profile on both switches is identical; otherwise, a merge-check failure might occur.

3. Configure the switches by entering the **sync-peers destination** command.
4. Verify that the switch profiles are the same by entering the appropriate **show** commands.

Isolating a Peer Switch

You can isolate a peer switch in order to make changes to a switch profile. This process can be used when you want to block configuration synchronization, debug configurations, or recover from a situation when the config-sync feature becomes out of sync.

Isolating a peer switch requires that you break the peer connection from the switch profile and then add the peer switch back to the switch profile.



Note For specific information on the following steps, see the appropriate sections in this chapter.

1. Remove the peer switch from the switch profile on both switches.
2. Add the **no sync-peers destination** command to the switch profile and commit the changes on both switches.
3. Add any necessary troubleshooting configurations.
4. Verify that the show running switch-profile is identical on both switches.
5. Add the **sync-peers destination ip-address** command to both switches and commit the changes.
6. Verify that the peers are in sync.

Deleting a Switch Profile

You can delete a switch profile.

Procedure

	Command or Action	Purpose
Step 1	config sync Example: <pre>switch# config sync switch(config-sync)#</pre>	Enters configuration synchronization mode.
Step 2	Required: no switch-profile name {all-config local-config} Example: <pre>switch(config-sync)# no switch-profile abc local-config switch(config-sync-sp)#</pre>	Deletes the switch profile as follows: • all-config—Deletes the switch profile on the local and the peer switch. If the peer switch is not reachable, only the local switch profile is deleted. • local-config—Deletes the switch profile and local configuration.

Note

	Command or Action	Purpose
		It is recommended that you execute resync-database prior to deleting a switch-profile: switch(config-sync)# resync-database
Step 3	(Optional) end Example: switch(config-sync-sp)# end switch#	Exits the switch profile configuration mode and returns to EXEC mode.
Step 4	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration. When you enter this command, the config-sync feature triggers the same operation on the peer switch.

Manually Correcting Mutex and Merge Failures

You can manually correct mutex and merge failures when they occur.



Note If the conflict is on the peer switch, follow the steps in [Isolating a Peer Switch, on page 33](#) to correct the problem on that switch.

1. Import the offending command into the switch profile using the switch profile import mode.
2. Change the behavior as desired.

Verifying the Switch Profile Configuration

To display information about a switch profile, perform one of the following tasks:

Command	Purpose
show switch-profile <i>name</i>	Displays the commands in a switch profile.
show switch-profile <i>name</i> buffer	Displays the uncommitted commands in a switch profile, the commands that were moved, and the commands that were deleted.
show switch-profile <i>name</i> peer <i>ip-address</i>	Displays the synchronization status for a peer switch.
show switch-profile <i>name</i> session-history	Displays the status of the last 20 switch profile sessions.
show switch-profile <i>name</i> status	Displays the configuration synchronization status of a peer switch.

Command	Purpose
show running-config switch-profile	Displays the running configuration for the switch profile on the local switch.
show startup-config switch-profile	Displays the startup configuration for the switch profile on the local switch.

Configuration Examples for Switch Profiles

Creating a Switch Profile on a Local and a Peer Switch

The following example shows how to create a successful switch profile configuration on a local and a peer switch, including configuring QoS policies, a vPC peer link, and a vPC in a switch profile.

1. Enable CFS distribution on the local and the peer switch and configure the destination IP address of the switch that you want to synchronize with, such as the management interface on the switch.

```
-Local switch-1#---
switch-1# configure terminal
switch-1(config)# cfs ipv4 distribute
switch-1(config)# interface mgmt 0
switch-1(config-if)# ip address 30.0.0.81/8
```

```
-Peer switch-2#--
switch-2# configure terminal
switch-2(config)# cfs ipv4 distribute
switch-2(config)# interface mgmt 0
switch-2(config-if)# ip address 30.0.0.82/8
```

2. Create a new switch profile on the local and the peer switch.

```
-Local switch-1#---
switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# sync-peers destination 30.0.0.82
switch-1(config-sync-sp)# end
```

```
-Peer switch-2#--
switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# sync-peers destination 30.0.0.81
switch-1(config-sync-sp)# end
```

3. Verify that the switch profiles are the same on the local and the peer switch.

```
switch-1(config-sync-sp)# show switch-profile status
```

```
switch-profile : A
```

```
-----
Start-time: 843992 usecs after Wed Aug 19 17:00:01 2015
End-time: 770051 usecs after Wed Aug 19 17:00:03 2015
```

```

Profile-Revision: 1
Session-type: Initial-Exchange
Session-subtype: Init-Exchange-All
Peer-triggered: Yes
Profile-status: Sync Success

```

Local information:

Status: Commit Success

Error(s):

Peer information:

IP-address: 30.0.0.82

Sync-status: In sync

Status: Commit Success

Error(s):

4. Add the configuration commands to the switch profile on the local switch. The commands will be applied to the peer switch when the commands are committed.

```

switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface port-channel 10
switch-1(config-sync-sp-if)# switchport
switch-1(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface port-channel 10
switch-1(config-sync-sp-if)# switchport mode trunk
switch-1(config-sync-sp-if)# switchport trunk allowed vlan 10
switch-1(config-sync-sp-if)# spanning-tree port type network
switch-1(config-sync-sp-if)# vpc peer-link
switch-1(config-sync-sp-if)# switch-profile switching-mode switchname
switch-1(config-sync-sp-if)# show switch-profile buffer

switch-profile : A
-----
Seq-no Command
-----

1 interface port-channell0
1.1 switchport mode trunk
1.2 switchport trunk allowed vlan 10
1.3 spanning-tree port type network
1.4 vpc peer-link

switch-1(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface ethernet 2/1
switch-1(config-sync-sp-if)# switchport mode trunk

```

```
switch-1(config-sync-sp-if)# switchport trunk allowed vlan 10
switch-1(config-sync-sp-if)# spanning-tree port type network
switch-1(config-sync-sp-if)# channel-group 10 mode active
```

5. View the buffered commands.

```
switch-1(config-sync-sp-if)# show switch-profile buffer

switch-profile : A
-----
Seq-no Command
-----
1 interface Ethernet2/1
1.1 switchport mode trunk
1.2 switchport trunk allowed vlan 10
1.3 spanning-tree port type network
1.4 channel-group 10 mode active
```

6. Verify the commands in the switch profile.

```
switch-1(config-sync-sp-if)# verify
Verification Successful
```

7. Apply the commands to the switch profile and synchronize the configurations between the local and the peer switch.

```
-Local switch-2#--
switch-1(config-sync-sp)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch-1(config-sync)# end

switch-1# show running-config switch-profile

switch-profile A
sync-peers destination 30.0.0.82

interface port-channel10
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
vpc peer-link

interface Ethernet2/1
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
channel-group 10 mode active

-Peer switch-2#--
switch-2# show running-config switch-profile

switch-profile A
sync-peers destination 30.0.0.81

interface port-channel10
switchport mode trunk
switchport trunk allowed vlan 10
```

```

spanning-tree port type network
vpc peer-link

interface Ethernet2/1
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
channel-group 10 mode active

```

Verifying the Synchronization Status

The following example shows how to verify the synchronization status between the local and the peer switch:

```

switch-1# show switch-profile status

switch-profile : A
-----switch-1-----

Start-time: 912776 usecs after Wed Aug 19 17:03:43 2015
End-time: 868379 usecs after Wed Aug 19 17:03:48 2015

Profile-Revision: 4
Session-type: Commit
Session-subtype: -
Peer-triggered: No
Profile-status: Sync Success

Local information:
-----
Status: Commit Success
Error(s):

Peer information:
-----
IP-address: 30.0.0.82
Sync-status: In sync
Status: Commit Success
Error(s):

```

Showing the Running Configuration

The following example shows the running configuration of the switch profile on the local switch:

```

—— PEER SWITCH-1 ——
switch-1# show running-config switch-profile

switch-profile A
sync-peers destination 30.0.0.82

interface port-channel10
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
vpc peer-link

interface Ethernet2/1
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network

```

```
channel-group 10 mode active
switch-1#

— PEER SWITCH-2 —
switch-2# show running-config switch-profile

switch-profile A
sync-peers destination 30.0.0.81

interface port-channel10
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
vpc peer-link

interface Ethernet2/1
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
channel-group 10 mode active
switch-2#
```

Displaying the Switch Profile Synchronization Between the Local and the Peer Switch

The following example shows how to display the initial successful synchronization between the two peers:

```
switch1# show switch-profile sp status

Start-time: 491815 usecs after Mon Jul 20 11:54:51 2015
End-time: 449475 usecs after Mon Jul 20 11:54:58 2015

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: No
Profile-status: Sync Success

Local information:
-----
Status: Commit Success
Error(s):

Peer information:
-----
IP-address: 10.193.194.52
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch2# show switch-profile sp status

Start-time: 503194 usecs after Mon Jul 20 11:54:51 2015
End-time: 532989 usecs after Mon Jul 20 11:54:58 2015

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
-----
```

```

Status: Commit Success
Error(s):

Peer information:
-----
IP-address: 10.193.194.51
Sync-status: In Sync.
Status: Commit Success
Error(s):

```

Displaying Verify and Commit on the Local and the Peer Switch

The following example shows how to perform a successful verify and commit of the local and the peer switch:

```

switch1# config sync
switch1(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch1(config-sync-sp)# interface Ethernet1/1
switch1(config-sync-sp-if)# description foo
switch1(config-sync-sp-if)# exit
switch1(config-sync-sp)# verify
Verification Successful
switch1(config-sync-sp)# commit
Commit Successful
switch1(config-sync)# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.52
  interface Ethernet1/1
    description foo
switch1(config-sync)# show switch-profile sp status

Start-time: 171513 usecs after Wed Jul 20 17:51:28 2015
End-time: 676451 usecs after Wed Jul 20 17:51:43 2015

Profile-Revision: 3
Session-type: Commit
Peer-triggered: No
Profile-status: Sync Success

Local information:
-----
Status: Commit Success
Error(s):

Peer information:
-----
IP-address: 10.193.194.52
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch1(config-sync)#

switch2# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.51
  interface Ethernet1/1
    description foo
switch2# show switch-profile sp status

```

```

Start-time: 265716 usecs after Mon Jul 20 16:51:28 2015
End-time: 734702 usecs after Mon Jul 20 16:51:43 2015

Profile-Revision: 3
Session-type: Commit
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
-----
Status: Commit Success
Error(s):

Peer information:
-----
IP-address: 10.193.194.51
Sync-status: In Sync.
Status: Commit Success
Error(s):

```

Displaying the Successful and Unsuccessful Synchronization Between the Local and the Peer Switch

The following example shows how to configure the synchronization status of the switch profile on the peer switch. The first example shows a successful synchronization, and the second example shows a peer-not-reachable status.

```

switch1# show switch-profile sp peer

switch1# show switch-profile sp peer 10.193.194.52
Peer-sync-status      : In Sync.
Peer-status           : Commit Success
Peer-error(s)         :
switch1#

switch1# show switch-profile sp peer 10.193.194.52
Peer-sync-status      : Not yet merged. pending-merge:1 received_merge:0
Peer-status           : Peer not reachable
Peer-error(s)         :

```

Displaying the Switch Profile Buffer

The following example shows how to configure the switch profile buffer, the buffer-move configuration, and the buffer-delete configuration:

```

switch1# config sync
switch1(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch1(config-sync-sp)# vlan 101
switch1(config-sync-sp-vlan)# ip igmp snooping querier 10.101.1.1
switch1(config-sync-sp-vlan)# exit
switch1(config-sync-sp)# mac address-table static 0000.0000.0001 vlan 101 drop
switch1(config-sync-sp)# interface Ethernet1/2
switch1(config-sync-sp-if)# switchport mode trunk
switch1(config-sync-sp-if)# switchport trunk allowed vlan 101
switch1(config-sync-sp-if)# exit

```

```

switch1(config-sync-sp)# show switch-profile sp buffer
-----
Seq-no  Command
-----
1       vlan 101
1.1     ip igmp snooping querier 10.101.1.1
2       mac address-table static 0000.0000.0001 vlan 101 drop
3       interface Ethernet1/2
3.1     switchport mode trunk
3.2     switchport trunk allowed vlan 101

switch1(config-sync-sp)# buffer-move 3 1
switch1(config-sync-sp)# show switch-profile sp buffer
-----
Seq-no  Command
-----
1       interface Ethernet1/2
1.1     switchport mode trunk
1.2     switchport trunk allowed vlan 101
2       vlan 101
2.1     ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop

switch1(config-sync-sp)# buffer-delete 1
switch1(config-sync-sp)# show switch-profile sp buffer
-----
Seq-no  Command
-----
2       vlan 101
2.1     ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop

switch1(config-sync-sp)# buffer-delete all
switch1(config-sync-sp)# show switch-profile sp buffer

```

Importing Configurations

The following example shows how to import an interface configuration:

```

switch# show running-config interface Ethernet1/3

!Command: show running-config interface Ethernet1/3
!Time: Wed Jul 20 18:12:44 2015

version 7.0(3)I2(1)

interface Ethernet1/3
  switchport mode trunk
  switchport trunk allowed vlan 1-100

switch# config sync
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1

switch(config-sync-sp)# import interface Ethernet1/3
switch(config-sync-sp-import)# show switch-profile sp buffer
-----
Seq-no  Command
-----
1       interface Ethernet1/3
1.1     switchport mode trunk

```



```
1.2      switchport trunk allowed vlan 1-100
```

```
switch(config-sync-sp-import)# verify
Verification Successful
switch(config-sync-sp-import)# commit
Commit Successful
```

The following example shows how to import the supported commands in a running configuration:

```
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# import running-config
switch(config-sync-sp-import)# show switch-profile sp buffer
-----
Seq-no  Command
-----
1       logging event link-status default
2       vlan 1
3       interface port-channel 3
3.1     switchport mode trunk
3.2     vpc peer-link
3.3     spanning-tree port type network
4       interface port-channel 30
4.1     switchport mode trunk
4.2     vpc 30
4.3     switchport trunk allowed vlan 2-10
5       interface port-channel 31
5.1     switchport mode trunk
5.2     vpc 31
5.3     switchport trunk allowed vlan 11-20
6       interface port-channel 101
6.1     switchport mode fex-fabric
6.2     fex associate 101
7       interface port-channel 102
7.1     switchport mode fex-fabric
7.2     vpc 102
7.3     fex associate 102
8       interface port-channel 103
8.1     switchport mode fex-fabric
8.2     vpc 103
8.3     fex associate 103
9       interface Ethernet1/1
10      interface Ethernet1/2
11      interface Ethernet1/3
12      interface Ethernet1/4
12.1    switchport mode trunk
12.2    channel-group 3
13      interface Ethernet1/5
13.1    switchport mode trunk
13.2    channel-group 3
14      interface Ethernet1/6
14.1    switchport mode trunk
14.2    channel-group 3
15      interface Ethernet1/7
15.1    switchport mode trunk
15.2    channel-group 3
16      interface Ethernet1/8
17      interface Ethernet1/9
17.1    switchport mode trunk
17.2    switchport trunk allowed vlan 11-20
17.3    channel-group 31 mode active
18      interface Ethernet1/10
18.1    switchport mode trunk
18.2    switchport trunk allowed vlan 11-20
```

```

18.3      channel-group 31 mode active
19        interface Ethernet1/11
20        interface Ethernet1/12
...
45        interface Ethernet2/4
45.1      fex associate 101
45.2      switchport mode fex-fabric
45.3      channel-group 101
46        interface Ethernet2/5
46.1      fex associate 101
46.2      switchport mode fex-fabric
46.3      channel-group 101
47        interface Ethernet2/6
47.1      fex associate 101
47.2      switchport mode fex-fabric
47.3      channel-group 101
48        interface Ethernet2/7
48.1      fex associate 101
48.2      switchport mode fex-fabric
48.3      channel-group 101
49        interface Ethernet2/8
49.1      fex associate 101
...
89        interface Ethernet100/1/32
90        interface Ethernet100/1/33
91        interface Ethernet100/1/34
92        interface Ethernet100/1/35
93        interface Ethernet100/1/36
...
105       interface Ethernet100/1/48

```

Migrating to Cisco NX-OS Release 7.0(3)I2(1) or Higher in a Fabric Extender Straight-Through Topology

This example shows the tasks used to migrate to Cisco NX-OS Release 7.0(3)I2(1) or higher in a Fabric Extender active/active or straight-through topology. For details on the tasks, see the appropriate sections in this chapter.

1. Make sure configurations are the same on both switches.
2. Configure the switch profile with the same name on both switches.
3. Enter the **import interface port-channel** *x-y*, **port-channel** *z* command for all vPC port channels on both switches.
4. Enter the **show switch-profile** *name* **buffer** command to ensure all configurations are correctly imported on both switches.
5. Remove unwanted configuration settings by editing the buffer.
6. Enter the **commit** command on both switches.
7. Enter the **sync-peers destination** *ip-address* command to configure the peer switch on both switches.
8. Enter the **show switch-profile** *name* **status** command to ensure both switches are synchronized.

Replacing a Cisco Nexus 9000 Series Switch

When a Cisco Nexus 9000 Series switch has been replaced, perform the following configuration steps on the replacement switch to synchronize it with the existing Cisco Nexus 9000 Series switch. This procedure can be done in a hybrid Fabric Extender active/active topology and Fabric Extender straight-through topology.

1. Do not connect any peer link, vPC, active/active, or straight-through topology fabric ports to the replacement switch.
2. Boot the replacement switch. The switch comes up with no configuration.
3. Configure the replacement switch:
 - If the running configuration was saved offline, follow Steps 4 through 8 to apply the configuration.
 - If the running configuration was not saved offline, you can obtain it from the peer switch if the configuration synchronization feature is enabled. (See Steps 1 and 2 in [Creating a Switch Profile on a Local and a Peer Switch, on page 35](#); then begin with Step 9 below).
 - If neither condition is met, manually add the configuration and then begin with Step 9 below.
4. Edit the configuration file to remove the **sync-peer** command if you are using the configuration synchronization feature.
5. Configure the mgmt port IP address and download the configuration file.
6. Copy the saved configuration file to the running configuration.
7. Verify that the configuration is correct by entering the **show running-config** command.
8. If the switch profile configuration changes were made on the peer switch while the replacement switch was out of service, apply those configurations in the switch profile and then enter the **commit** command.
9. Shut down all Fabric Extender straight-through topology ports that are included in a vPC topology.
10. Connect the Fabric Extender straight-through topology fabric ports.
11. Wait for the Fabric Extender straight-through topology switches to come online.
12. Make sure that the vPC role priority of the existing switch is better than the replacement switch.
13. Connect the peer-link ports to the peer switch.
14. Connect the switch vPC ports.
15. Enter the **no shutdown** command on all Fabric Extender straight-through vPC ports.
16. Verify that all vPC switches and the Fabric Extenders on the replacement switch come online and that there is no disruption in traffic.
17. If you are using the configuration synchronization feature, add the sync-peer configuration to the switch profile if it was not enabled in Step 3.
18. If you are using the configuration synchronization feature, enter the **show switch-profile name status** command to ensure both switches are synchronized.

Synchronizing Configurations

Synchronizing Configurations After a Cisco Nexus 9000 Series Switch Reboots

If a Cisco Nexus 9000 Series switch reboots while a new configuration is committed on a peer switch using a switch profile, follow these steps to synchronize the peer switches after the reload:

1. Remove the peer switch from the switch profile on both switches.
2. Add the **no sync-peers destination** command to the switch profile and commit the changes on both switches.
3. Add any missing or changed commands.
4. Verify that the show running switch-profile is identical on both switches.
5. Add the **sync-peers destination ip-address** command to both switches and commit the changes.
6. Verify that the peers are in sync.

Synchronizing Configurations When the mgmt0 Interface Connectivity Is Lost

When the mgmt0 interface connectivity is lost and configuration changes are required, apply the configuration changes on both switches using the switch profile. When connectivity to the mgmt0 interface is restored, both switches are synchronized.

If a configuration change is made on only one switch in this scenario, a merge will succeed when the mgmt0 interface comes up and the configuration gets applied on the other switch.

Reverting an Inadvertent Port Mode Change of Layer 2 to Layer 3 in Global Configuration Mode

The configurations related to a port imported in config-sync mode should never be configured in the global configuration mode. Normally any attempt to do so will be denied by the config-sync feature, and a mutex warning will appear. However, due to limitations in mutex checks, if a port configured as Layer 2 in the config-sync mode is changed to Layer 3 (no switchport) in the global configuration mode, the config-sync feature is unable to detect and prevent it. As a result, the config-sync mode might become out of sync with the global configuration mode. In this case, follow these steps to revert the change:

1. Remove the peer switch from the switch profile on both switches.
2. Add the **no sync-peers destination** command to the switch profile and commit the changes on both switches.
3. Import the current interface configuration.
4. Make any necessary changes and commit them.
5. Verify that the show running switch-profile is identical on both switches.
6. Add the **sync-peers destination ip-address** command to both switches and commit the changes.
7. Verify that the peers are in sync.



CHAPTER 5

Configuring Frequency Synchronization

This chapter describes how to configure Frequency Synchronization on Cisco NX-OS devices.

This chapter includes the following sections:

- [About Frequency Synchronization, on page 47](#)
- [Licensing Requirements for Synchronous Ethernet \(SyncE\), on page 50](#)
- [Guidelines and Limitations for Frequency Synchronization, on page 50](#)
- [Configuring Frequency Synchronization, on page 51](#)

About Frequency Synchronization

Next generation networks must provide the ability to distribute precision frequency around the network. This is known as frequency synchronization. Precision frequency is required for applications such as circuit emulation and cell tower frequency referring. To achieve compliance to ITU specifications for TDM, differential method circuit emulation must be used, which requires a known, common precision frequency reference at each end of the emulated circuit.

It is also often desirable to precisely synchronize the time-of-day between different network devices, for example in order to accurately calculate the packet delay between two nodes in the network.

As, increasingly, SDH and SONET equipment is replaced by Ethernet equipment, this frequency synchronization ability is required over Ethernet ports. Synchronous Ethernet (SyncE) provides this PHY-level frequency distribution of known common precision frequency references.

To maintain SyncE links, a set of operations messages are required. These messages ensure a node is always deriving timing from the most reliable source, and transfer information about the quality of the timing source being used to clock the SyncE link. A simple protocol providing a transport channel for Synchronization Status Messages (SSMs) over Ethernet is documented in the ITU standard G.8264 and its related recommendations.

Each timing source has a Quality Level (QL) associated with it which gives the accuracy of the clock. This QL information is transmitted across the network via SSMs over the Ethernet Synchronization Messaging Channel (ESMC) so that devices can know the best available source to use for synchronization. In order to define a preferred network synchronization flow, and to help prevent timing loops, priority values can be assigned to particular timing sources on each switch. The combination of QL information and user-assigned priority levels allows each switch to choose a timing source to use to clock its SyncE as described in the ITU standard G.781.

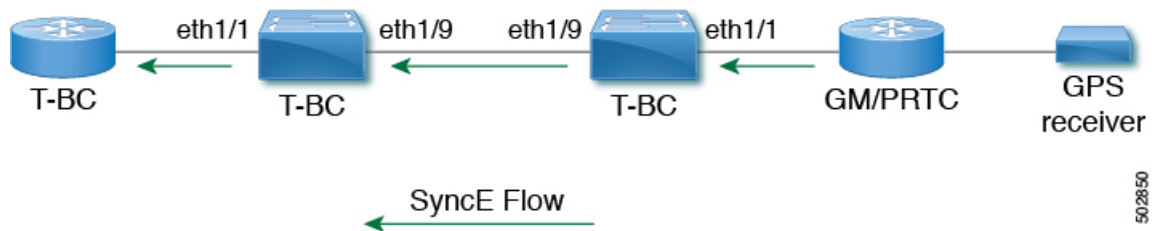
SyncE does not carry time-of-day information. Time-of-day synchronization is achieved using packet-based technologies, such as PTP. Clock sources such as GNSS/GPS can be used to inject accurate time-of-day, as well as frequency, into the network. Each switch in the network can select a source for time-of-day as well as a source for frequency (or select the same source for both, if possible and desirable), and pass its time-of-day information to its peers using a packet-based protocol. There is no equivalent to QL for time-of-day information, so selection between different sources for time-of-day is achieved using configuration.

Hybrid SyncE-PTP with External PRC Source

Beginning with Cisco NX-OS Release 9.3(5), a hybrid SyncE-PTP topology is supported to achieve the end-to-end network precision frequency required for circuit emulation and cell tower frequency referring.

The following figure shows the external timing source as the Grandmaster/Primary Reference Time Clock (GM/PRTC) providing the timing source for the Telecom Boundary Clocks (T-BCs).

Figure 3: Hybrid SyncE-PTP with External PRC Source



Timing Sources

There are various timing sources that input timing clock signals into the system/network, and outputs of timing clock signals from the system as described below.

Timing Inputs

Input clock signals can be received from the platform hardware either via inputs from timing sources like GPS/GNSS, from internal oscillators, recovered from the line of a SyncE enabled interface, or from timing over packet such as the Precision Time Protocol (PTP).

Platform independent (PI) software keeps a database of all these inputs, including a Quality Level (QL) and priority level associated with each. The priority level is configuration controlled, and the QL values can be obtained in a number of manners:

- SyncE enabled interfaces receive SSMs via an Ethernet slow protocol (ESMC).
- GPS and GNSS have fixed QL maintained by platform dependent (PD) software and notified to PI function.
- PTP communicates its QL to the Frequency Synchronization PI software through the platform APIs.
- A default QL value may be defined in the PD layer for the timing connectors, and internal oscillators.
- Configuration may be set defining the QLs of timing sources.

Possible input sources:

- Internal Oscillator

- Recovered SyncE Clock
- External Clock 1588/PTP
- External Clock (GPS)
- Internal Clock (GNSS)

Timing Outputs

The platform hardware can have a number of outputs for clock signals like timing clock outputs from SyncE and enabled interfaces for GPS (currently not supported).

The software keeps all these outputs in a database, including QL information associated with the clock signal being used to drive these outputs that may be explicitly configured. The QL information includes a QL value along with steps removed counters, the originator clock ID and a series of flags containing information about the path from the originator clock to the current clock. The QL values are transmitted in the same manners as described for the inputs (i.e. SyncE interfaces send ESMC SSMs).

Possible output sources:

- SyncE
- 1588/PTP: packet output is handled separately, in the PTP software.

Timing Source Selection Points

At various stages in syncing timing clocks around the system, the platform has the potential to make a choice over which of the available timing clocks it is to use for further processing. These selection points define the flow of timing clock signals through the system, and eventually lead to the overall decision on which input timing source is to be used for timing outputs.

How these selection points are setup on each platform is hardware dependent, but the platform independent (PI) layer defines a generic selection point abstraction that can flexibly represent any platform selection point hardware, and allows each platform to define which selection points it has, and how they are wired together. The PI code can then control these selection points, tracking and distributing required information about the timing sources, and interacting with the platform dependent (PD) layer to discover what the result of the PD selection is at each stage.

PI timing source selection points:

- Available Timing Inputs: A number of timing clock inputs are available for the platform selection point hardware to choose between. The availability and associated QL information and priorities are tracked by PI software, which informs the PD layer which inputs are available, ranked in overall order along with their associated quality levels and priorities.
- Platform Specific Selection: The platform layer makes a decision as to which of the inputs it is using based on the information obtained from PI, and other platform layer decisions (e.g. hardware level qualification of the clock-signals). The actual decision may be made in PD software (and programmed into the hardware), or the decision may be made by the hardware itself and communicated back to the PD software.
- Selected Timing Source Outputs: The platform passes the selected clock signal(s) through as output(s) from the selection point. The PD layer informs the PI software the status of the available inputs, and which input(s) have been selected.

The platform layer defines what the selection points are, and how they are connected to potential inputs, and to each other, and to potential outputs. At each of the PD defined selection points, the platform can choose how to interact with the PI software to represent its particular hardware to the PI software. The hardware doesn't have to perform clocking qualification at each selection point. Each selection point simply represents any place where the hardware selects between multiple inputs, passing the clock from one or many inputs forward.

Only one selection point type for SyncE on the switch supervisor is supported. This is named T0 and 1588 selection points. The T0 selection point represents the sources and its selection for the SyncE DPLL. The 1588 selection point represents the sources and its selection for the Assist DPLL for 1588 PLL.

Licensing Requirements for Synchronous Ethernet (SyncE)

Product	License Requirement
Cisco NX-OS	SyncE requires an add-on license. For a complete explanation of the Cisco NX-OS licensing Cisco NX-OS Licensing Guide .

Guidelines and Limitations for Frequency Synchronization

Frequency Synchronization has the following guidelines and limitations:

- Refer to [Nexus Switch Platform Support Matrix](#) to see the list of Cisco Nexus switches that support the Frequency Synchronization (SyncE) feature through Cisco NX-OS releases.
- SyncE is supported only on physical interfaces.
- A maximum four ethernet interfaces can be monitored for SyncE selection input at any given instance of time.
- Each quad port group on the PHY provides one reference clock.
- Only one Ethernet interface from each quad port group can be configured as a SyncE input (one reference clock for each port group). There is no restriction on SyncE outputs.
- SyncE must be enabled explicitly on the member interfaces for a port-channel. If a member interface of a port-channel is locked as a SyncE source, the ability to send out DNU on other member interfaces enabled for SyncE is controlled via the global command **fsync transmit dnu lag-members**.
- Only G.8275.1 hybrid profile in BC mode is supported.
- For a list of qualified optics for this release, see the [Cisco Optics Compatibility Matrix](#).



Note SyncE is not supported on 1G when GLC-TE is used as SFP.

- Beginning with Cisco NX-OS Release 10.3(2)F, GPS and GNSS is supported on the Cisco Nexus 93180YC-FX3 switch.

Configuring Frequency Synchronization

Enabling Frequency Synchronization

Use this procedure to enable frequency synchronization, set the quality level of the switch, identify the clock ID for ESMC extended TLV, and configure the ESMC peer timeout for software upgrades.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] feature frequency-synchronization Example: <pre>switch(config)# feature frequency-synchronization switch(config)#</pre>	Enables frequency synchronization on the switch.
Step 3	[no] fsync quality itu-t option { 1 2 generation { 1 2 } Example: <pre>switch(config)# fsync quality itu-t option 1 switch(config)#</pre>	Specifies the quality level for the switch. The default is option 1. • option 1 - Includes DNU, EEC1, PRC, PRTC, SEC, SSU-A, SSU-B, eEEC and ePRTC. • option 2 generation 1 - Includes DUS, EEC2, PRS, PRTC, RES, SMC, ST2, ST3, ST4, STU, eEEC and ePRTC. • option 2 generation 2 - Includes DUS, EEC2, PROV, PRS, PRTC, SMC, ST2, ST3, ST3E, ST4, STU, TNC, eEEC and ePRTC. Note The quality option that is configured here must match the quality option that is specified in the quality receive and quality transmit commands in the interface frequency synchronization configuration mode.
Step 4	fsync clock-identity mac-address no fsync clock-identity Example:	Specifies the clock ID to be used for Ethernet Synchronization Message Channel (ESMC) extended TLV. If no clock ID is configured, the system uses the default VDC MAC address.

	Command or Action	Purpose
	<pre>switch(config)# fsync clock-identity AB:CD:EF:12:34:56 switch(config)#</pre>	
Step 5	[no] fsync esmc peer receive timeout { 0 value } Example: <pre>switch(config)# fsync esmc peer receive timeout 120 switch(config)#</pre>	Specifies the ESMC peer receive timeout during ISSU. 0 disables the ESMC peer receive timeout. <i>value</i> is the ESMC receive timeout in seconds. Enter a value from 120 through 600. Default = 120. This command ensures that the ESMC control plane, and thus, selection, is not removed during software upgrade for a period of the <i>value</i>.
Step 6	[no] fsync transmit dnu lag-members Example: <pre>switch(config)# fsync transmit dnu lag-members switch(config)#</pre>	SyncE must be enabled explicitly on the member interfaces for a port-channel. If a member interface of a port-channel is locked as a SyncE source, the ability to send out DNU (Do Not Use) QLs on other member interfaces that are enabled for SyncE is controlled by this command. If enabled and an interface that is driving the clock for the switch is part of a port-channel, then any members of the port-channel will also send out DNU QL if SyncE is enabled on that interface. If disabled, the system drives the QL of the selected source on all interfaces regardless of whether they are in the same port-channel as the interface driving the clock.
Step 7	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config switch(config)#</pre>	Copies the running configuration to the startup configuration.

Configuring Frequency Synchronization on an Interface

Use this procedure to configure frequency synchronization on a specific interface.

Before you begin

This procedure, along with configuring PTP telecom profile on the same interface, constitutes the required interface settings for the "hybrid PTP" platform. For more information about the interface PTP telecom profile configuration, see [Configure PTP Telecom Profile 8275-1 on an Interface, on page 95](#).

Make sure that you have globally enabled frequency synchronization on the device (global configuration command **feature frequency-synchronization**).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] interface ethernet slot / port Example: <pre>switch(config)# interface ethernet 1/5 switch(config-if)#</pre>	Specifies the interface on which you are enabling frequency synchronization and enters the interface configuration mode.
Step 3	[no] frequency synchronization Example: <pre>switch(config-if)# frequency synchronization switch(config-if-freqsync)#</pre>	Enables frequency synchronization on the interface and enters the interface frequency synchronization configuration mode. The system selects the frequency signal to be used for clocking transmission, but does not enable the use of the interface as an input. Note The no form of the command functions only if there is no configuration present under the frequency synchronization configuration mode.
Step 4	[no] selection input Example: <pre>switch(config-if-freqsync)# selection input switch(config-if-freqsync)#</pre>	Specifies the interface as a timing source to be passed to the selection algorithm.
Step 5	[no] ssm disable Example: <pre>switch(config-if-freqsync)# ssm disable switch(config-if-freqsync)#</pre>	Disables sending ESMC packets and ignores any received ESMC packets.
Step 6	[no] quality { receive transmit } { exact highest lowest } itu-t option ql-option ql Example: <pre>switch(config-if-freqsync)# quality receive exact itu-t option 1 PRC switch(config-if-freqsync)#</pre>	Adjusts the Quality Level (QL) value that is used in received or transmitted SSMs, before it is used in the selection algorithm. Each timing source has a QL associated with it which provides the accuracy of the clock. This QL information is transmitted across the network via SSMs over the Ethernet Synchronization Messaging Channel (ESMC) so that devices can know the best available source to use for synchronization.

	Command or Action	Purpose
		• exact ql: Specifies the exact QL regardless of the value that is received, unless the received value is DNU. • highest ql: Specifies an upper limit on the received QL. If the received value is higher than this specified QL, this QL is used instead. • lowest ql: Specifies a lower limit on the received QL. If the received value is lower than this specified QL, DNU is used instead. Note The quality option that is specified in this command must match the globally configured quality option in the quality itu-t option command.
Step 7	[no] priority value Example: <pre>switch(config-if-freqsync)# priority 100 switch(config-if-freqsync)#</pre>	Configures the priority of the frequency source on the interface. This priority is used in the clock-selection algorithm to choose between two sources that have the same QL. Values range from 1 (highest priority) to 254 (lowest priority). The default value is 100. Note This command is valid only if selection input is configured.
Step 8	[no] wait-to-restore minutes Example: <pre>switch(config-if-freqsync)# wait-to-restore 0 switch(config-if-freqsync)#</pre>	Configures the wait-to-restore time, in minutes, for frequency synchronization on the interface. <i>minutes</i> is the amount of time after the interface initializes before it is used for synchronization. Values range from 0 to 12. The default value is 5. Note This command is valid only if selection input is configured.

Verifying the Frequency Synchronization Configuration

After performing the frequency synchronization configuration tasks, use this reference to check for configuration errors and verify the configuration.

show frequency synchronization configuration errors

The output of this command displays errors in the frequency synchronization configuration.

The following example shows the mismatch between the global **quality itu-t option** and the interface **quality receive itu-t option**:

```
switch# show frequency synchronization configuration errors

Elysian2(config)# show frequency synchronization configuration errors
Ethernet1/9
    quality receive exact itu-t option 1 PRC
* The QL that is configured is from a different QL option set than is
configured globally.

!Command: show running-config fsync_mgr all
!Running configuration last done at: Mon Feb 10 06:06:15 2020
!Time: Mon Feb 10 06:09:18 2020

version 9.3(5) Bios:version 00.04
feature frequency-synchronization

fsync quality itu-t option 2 generation 1 << must be the same as interface
fsync clock-identity 0
fsync esmc peer receive timeout 120

interface Ethernet1/9
    frequency synchronization
        selection input
        ssm disable
        quality receive exact itu-t option 1 PRC << must be the same as global
        priority 100
        wait-to-restore 0

interface Ethernet1/13
    frequency synchronization
        selection input
        ssm disable
        quality receive exact itu-t option 1 PRC
        priority 110
        wait-to-restore 0
```

show running-config fsync_mgr

The output of this command displays the current frequency synchronization configuration on the device.

The following is an example of the output of the **show running-config fsync_mgr** command:

```
switch# show running-config fsync_mgr

!Command: show running-config fsync_mgr
!Running configuration last done at: Mon Jun 29 13:49:34 2020
!Time: Mon Jun 29 13:50:51 2020

version 9.3(5) Bios:version 01.01
feature frequency-synchronization

interface Ethernet1/9
    frequency synchronization
        selection input
        priority 99
        wait-to-restore 0

interface Ethernet1/13
    frequency synchronization
        selection input
```

```

ssm disable
quality receive exact itu-t option 1 PRC
wait-to-restore 0

```

show frequency synchronization interface brief

The output of this command displays all interfaces that have frequency synchronization configured. Sources that have been nominated as inputs have 'S' in the Flags (Fl) column. Sources that have not been nominated as inputs do not have 'S' displayed.

The following is an example of the output of the **show frequency synchronization interface brief** command:

```

switch# show frequency synchronization interface brief

Flags:  > - Up           D - Down           S - Assigned for selection
         d - SSM Disabled x - Peer timed out i - Init state
         e - SSM Enabled  s - Output squelched
Fl      Interface      QLrcv QLuse Pri QLsnd Output driven by
=====
>S      Eth1/9          PRC   PRC   100 PRC   Eth1/13
>Sds    Eth1/13         n/a   PRC   100 n/a   Eth1/13

```

show frequency synchronization interface ethernet

The output of this command displays individual (user-selected) interfaces with associated frequency synchronization information.

The following is an example of the output of the **show frequency synchronization interface ethernet slot / port** command:

```

switch# show frequency synchronization interface ethernet 1/9

Interface State:UP
Assigned as input for Selection
Wait-to-restore time 0 minute(s)
SSM Enabled
Peer Up for 00:07:01, last SSM received 0.307s ago
Peer has come up 4 times and timed out 1 times
ESMC SSMs      Total Information      Event      DNU/DUS
Sent:          1097          1088          9          83
Received:      823          816          7          155
Input:
Up
Last received QL: PRC
Effective QL: PRC, Priority: 100
Originator clock ID: ffffffffefbfa543
SyncE steps: 1, eSyncE steps: 1
Not all steps run eSyncE; Chain of extended ESMC data is broken
Supports frequency
Output:
Selected source: Eth1/13
Selected source QL: PRC
Effective QL: PRC
Originator clock ID: ffffffffefbfa863
SyncE steps: 1, eSyncE steps: 1
Not all steps run eSyncE; Chain of extended ESMC data is broken
Next selection points:

```

show frequency synchronization selection (with PTP Profile 8275-1)

The output of this command displays the detailed view of the different selection points within the system.



Note This example shows the output when PTP profile 8275-1 is configured.

The following is an example of the output of the **show frequency synchronization selection slot / port** command:

```
switch# show frequency synchronization selection
=====
Selection point: System Clock (T0) Selector (3 inputs, 1 selected)
  Last programmed 18.898s ago, and selection made 8.621s ago
  Next selection points
    Node scoped :
  Uses frequency selection
  Used for local line interface output
  S  Input                               Last Selection Point      QL  Pri  Status
  == =====
  11 Ethernet1/9                        n/a                       PRC  99  Locked
      Ethernet1/13                      n/a                       PRC 100 Available
      Internal0[1]                      n/a                       SEC 255 Available
  =====
Selection point: IEEE 1588 Clock Selector (3 inputs, 1 selected)
  Last programmed 18.898s ago, and selection made 18.626s ago
  Next selection points
    Node scoped :
  Uses frequency selection
  S  Input                               Last Selection Point      QL  Pri  Status
  == =====
      Ethernet1/9                        n/a                       PRC  99  Unmonitored
      Ethernet1/13                      n/a                       PRC 100 Unmonitored
  21 Internal0[1]                      n/a                       SEC 255 Freerun  <<
```

show frequency synchronization selection (without PTP Profile 8275-1)

The output of this command displays the detailed view of the different selection points within the system.



Note This example shows the output when PTP profile 8275-1 is not configured.

The following is an example of the output of the **show frequency synchronization selection slot / port** command:

```
switch# show frequency synchronization selection=====
Selection point: System Clock (T0) Selector (3 inputs, 1 selected)
  Last programmed 00:03:04 ago, and selection made 00:02:54 ago
  Next selection points
    Node scoped :
  Uses frequency selection
  Used for local line interface output
  S  Input                               Last Selection Point      QL  Pri  Status
  == =====
  11 Ethernet1/9                        n/a                       PRC  99  Locked
      Ethernet1/13                      n/a                       PRC 100 Available
      Internal0[1]                      n/a                       SEC 255 Available
  =====
```

```

Selection point: IEEE 1588 Clock Selector (3 inputs, 1 selected)
Last programmed 00:03:04 ago, and selection made 3.296s ago
Next selection points
Node scoped      :
Uses frequency selection
S  Input          Last Selection Point      QL  Pri  Status
== =====
Ethernet1/9       n/a                    PRC  99  Unmonitored
Ethernet1/13      n/a                    PRC 100 Unmonitored
21 Internal0[1]   n/a                    SEC 255 Holdover  <<

```

show esmc counters all

The output of this command displays counters for sent and received ESMC SSMs.

The following is an example of the output of the **show esmc counters all** command:

```

ESMC Packet Counters of Interface Ethernet1/1:
ESMC SSMs      Total  Information  Event      DNU/DUS
Sent:          0          0          0          0
Received:      0          0          0          0

ESMC Packet Counters of Interface Ethernet1/5:
ESMC SSMs      Total  Information  Event      DNU/DUS
Sent:          0          0          0          0
Received:      0          0          0          0

ESMC Packet Counters of Interface Ethernet1/9:
ESMC SSMs      Total  Information  Event      DNU/DUS
Sent:          7685      7683          2          0
Received:      7688      7682          6          19

```

show esmc counters interface ethernet

The output of this command displays counters for sent and received ESMC SSMs on a specific interface.

The following is an example of the output of the **show esmc counters interface ethernet slot / port** command:

```

ESMC Packet Counters of Interface Ethernet1/9:
ESMC SSMs      Total  Information  Event      DNU/DUS
Sent:          7955      7953          2          0
Received:      7958      7952          6          19

```




CHAPTER 6

Precision Time Protocol

Precision Time Protocol (PTP) is a time synchronization protocol defined in IEEE 1588 for nodes distributed across a network. With PTP, it is possible to synchronize distributed clocks with an accuracy of less than 1 microsecond via Ethernet networks. In addition, PTP's hardware timestamping feature provides timestamp information in the ERSPAN Type III header that can be used to calculate packet latency among edge, aggregate, and core switches.

PTP is a distributed protocol that specifies how real-time PTP clocks in the system synchronize with each other. These clocks are organized into a master-slave synchronization hierarchy with the grandmaster clock, which is the clock at the top of the hierarchy, determining the reference time for the entire system. Synchronization is achieved by exchanging PTP timing messages, with the members using the timing information to adjust their clocks to the time of their master in the hierarchy. PTP operates within a logical scope called a PTP domain.

A PTP system can consist of a combination of PTP and non-PTP devices. PTP devices include ordinary clocks, boundary clocks, and transparent clocks. Non-PTP devices include ordinary network switches, routers, and other infrastructure devices.

- [Functionality, on page 59](#)
- [Guidelines and limitations for PTP, on page 67](#)
- [Default PTP Parameters, on page 74](#)
- [Configure PTP, on page 75](#)
- [Enable PTP Unicast-Negotiation, on page 106](#)
- [Enhanced Multicast Scale, on page 109](#)
- [Timestamp Tagging, on page 109](#)
- [Verify PTP Configuration, on page 113](#)
- [Configuration Examples, on page 118](#)
- [Additional References, on page 120](#)

Functionality

PTP supports the following functionality:

- Multicast and unicast PTP transport—In the multicast transport mode, PTP uses multicast destination IP address 224.0.1.129 as per IEEE 1588 standards for communication between devices. For the source IP address, it uses the user configurable global IP address under the PTP domain. In the unicast transport mode, PTP uses configurable unicast source and destination IP addresses that can be configured under

an interface. In both, the unicast and the multicast modes, PTP uses UDP ports, 319 for event messages and 320 for general messages communication between devices.

- PTP multicast configuration is supported only under physical interface for L2 or L3. Unicast PTP configuration supported only under L3 physical interface. PTP is not supported for virtual interfaces such as Port-channel, SVI, and tunnel.
- PTP encapsulation over UDP over IP—PTP uses UDP as the transport protocol over IP. In both, the unicast and multicast modes, PTP uses UDP ports 319 for event messages and 320 for general messages communication between devices. L2 encapsulation mode is not supported.
- PTP profiles—PTP supports default (1588), AES67, and SMPTE 2059-2 profiles. They all have different ranges of sync and delay request intervals. For information on the default profile, refer to IEEE 1588. For more information on AES67 and SMPTE 2059-2, refer to the respective specifications.
- Path delay measurement—We support delay request and response mechanism to measure the delay between the master and slave devices. Peer delay request and response mechanism is not supported.
- Message intervals—You can configure the interval at which the announce, sync, and delay request messages need to be sent between devices.
- Best master clock (BMC) selection—BMC algorithm is used to select master, slave, and passive states of the PTP enabled interfaces based on the Announce message received as per 1588 specification.
- PTP Offload - This feature distributes the PTP functionality to the line cards and allows scaling of the number of PTP sessions that are supported on the system. This feature is available for Cisco Nexus 9500 platform switches with N9K-X97160YC-EX line card, 9700-FX, 9636C-R, 9636Q-R, 9624D-R2, and 9636C-RX line cards.
- High Availability - Stateful restarts are not supported for PTP. After a reboot or a supervisor switchover, the running configuration is applied. For more information on high availability, see the [Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide](#).
- Mixed Mode - PTP supports Mixed mode for delivering PTP messages, which is detected automatically by Cisco Nexus device, based on the type of **delay_req** message received from connected client and no configuration is required. In this mode when slave sends **delay_req** in unicast message, master also replies with unicast **delay_resp** message.

PTP Offload

This feature distributes the PTP functionality to the line cards and allows scaling of the number of PTP sessions that are supported on the system. This feature is available for Cisco Nexus 9500 platform switches with 9700-EX, 9700-FX, 9636C-R, 9636Q-R, 9624D-R2, and 9636C-RX line cards.

Clocks

Clock Types

The PTP device type is configurable and can be used to set the clock type. The following clocks are common PTP devices:

- Ordinary clock - Communicates with the network based on a single physical port, similar to an end host. An ordinary clock can function as a grandmaster clock.

- Boundary clock - Typically has several physical ports, with each port behaving like a port of an ordinary clock. However, each port shares the local clock, and the clock data sets are common to all ports. Each port decides its individual state, either master (synchronizing other ports connected to it) or slave (synchronizing to a downstream port), based on the best clock available to it through all of the other ports on the boundary clock. Messages related to synchronization and establishing the master-slave hierarchy terminate in the protocol engine of a boundary clock and are not forwarded.
- Transparent clock - Forwards all PTP messages like an ordinary switch or router but measures the residence time of a packet in the switch (the time that the packet takes to traverse the transparent clock) and in some cases the link delay of the ingress port for the packet. The ports have no state because the transparent clock does not need to synchronize to the grandmaster clock. There are two kinds of transparent clocks:
 - End-to-end transparent clock - Measures the residence time of a PTP message and accumulates the times in the correction field of the PTP message or an associated follow-up message.
 - Peer-to-peer transparent clock - Measures the residence time of a PTP message and computes the link delay between each port and a similarly equipped port on another node that shares the link. For a packet, this incoming link delay is added to the residence time in the correction field of the PTP message or an associated follow-up message.
- Grandmaster clock - In a single PTP domain, the Grandmaster (GM) node acts as a primary clock source for the entire PTP network. The primary source for GM node will be taken either from internal GNSS system or external GNSS system. GM node can't synchronize time or frequency from other PTP node i.e. GM node can't have any slave ports, all ports act as only Master role.



Note PTP operates only in boundary clock mode. Cisco recommends deployment of a Grand Master Clock (10 MHz) upstream, with servers containing clocks requiring synchronization connected to the switch. End-to-end transparent clock and peer-to-peer transparent clock modes are not supported.

Clock Modes

The IEEE 1588 standard specifies two clock modes for the PTP supporting devices to operate in: one-step and two-step.

One-Step Mode:

In one-step mode the clock synchronization messages include the time at which the master port sends the message. The ASIC adds the timestamp to the synchronization message as it leaves the port. The master port operating in one-step mode is available for Cisco Nexus 9508-FM-R and 9504-FM-R fabric modules and Cisco Nexus 9636C-R, 9636Q-R, 9624D-R2, and 9636C-RX line cards.

The slave port uses the timestamp that comes as part of the synchronization messages.

Two-Step Mode:

In two-step mode the time at which the synchronization message leaves the port is sent in a subsequent follow-up message. This is the default mode.

Clocks

The following clocks are common PTP devices:

Ordinary clock

Communicates with the network based on a single physical port, similar to an end host. An ordinary clock can function as a grandmaster clock.

Boundary clock

Typically has several physical ports, with each port behaving like a port of an ordinary clock. However, each port shares the local clock, and the clock data sets are common to all ports. Each port decides its individual state, either master (synchronizing other ports connected to it) or slave (synchronizing to a downstream port), based on the best clock available to it through all of the other ports on the boundary clock. Messages related to synchronization and establishing the master-slave hierarchy terminate in the protocol engine of a boundary clock and are not forwarded.

Transparent clock

Forwards all PTP messages like an ordinary switch or router but measures the residence time of a packet in the switch (the time that the packet takes to traverse the transparent clock) and in some cases the link delay of the ingress port for the packet. The ports have no state because the transparent clock does not need to synchronize to the grandmaster clock.

There are two kinds of transparent clocks:

End-to-end transparent clock

Measures the residence time of a PTP message and accumulates the times in the correction field of the PTP message or an associated follow-up message.

Peer-to-peer transparent clock

Measures the residence time of a PTP message and computes the link delay between each port and a similarly equipped port on another node that shares the link. For a packet, this incoming link delay is added to the residence time in the correction field of the PTP message or an associated follow-up message.



Note PTP operates only in boundary clock mode. Cisco recommends deployment of a Grand Master Clock (10 MHz) upstream, with servers containing clocks requiring synchronization connected to the switch.

End-to-end transparent clock and peer-to-peer transparent clock modes are not supported.

Grandmaster clock

In a single PTP domain, the Grandmaster (GM) node acts as a primary clock source for the entire PTP network. The primary source for GM node will be taken either from internal GNSS system or external GNSS system. GM node can't synchronize time or frequency from other PTP node i.e. GM node can't have any slave ports, all ports act as only Master role.

Clock Modes

The IEEE 1588 standard specifies two clock modes for the PTP supporting devices to operate in: one-step and two-step.

One-Step Mode:

In one-step mode the clock synchronization messages include the time at which the master port sends the message. The ASIC adds the timestamp to the synchronization message as it leaves the port. The master port

operating in one-step mode is available for Cisco Nexus 9508-FM-R and 9504-FM-R fabric modules and Cisco Nexus 9636C-R, 9636Q-R, 9624D-R2, and 9636C-RX line cards.

The slave port uses the timestamp that comes as part of the synchronization messages.

Two-Step Mode:

In two-step mode the time at which the synchronization message leaves the port is sent in a subsequent follow-up message. This is the default mode.

Process

The PTP process consists of two phases: establishing the master-slave hierarchy and synchronizing the clocks.

Within a PTP domain, each port of an ordinary or boundary clock follows this process to determine its state:

- Examines the contents of all received announce messages (issued by ports in the master state)
- Compares the data sets of the foreign master (in the announce message) and the local clock for priority, clock class, accuracy, and so on
- Determines its own state as either master or slave

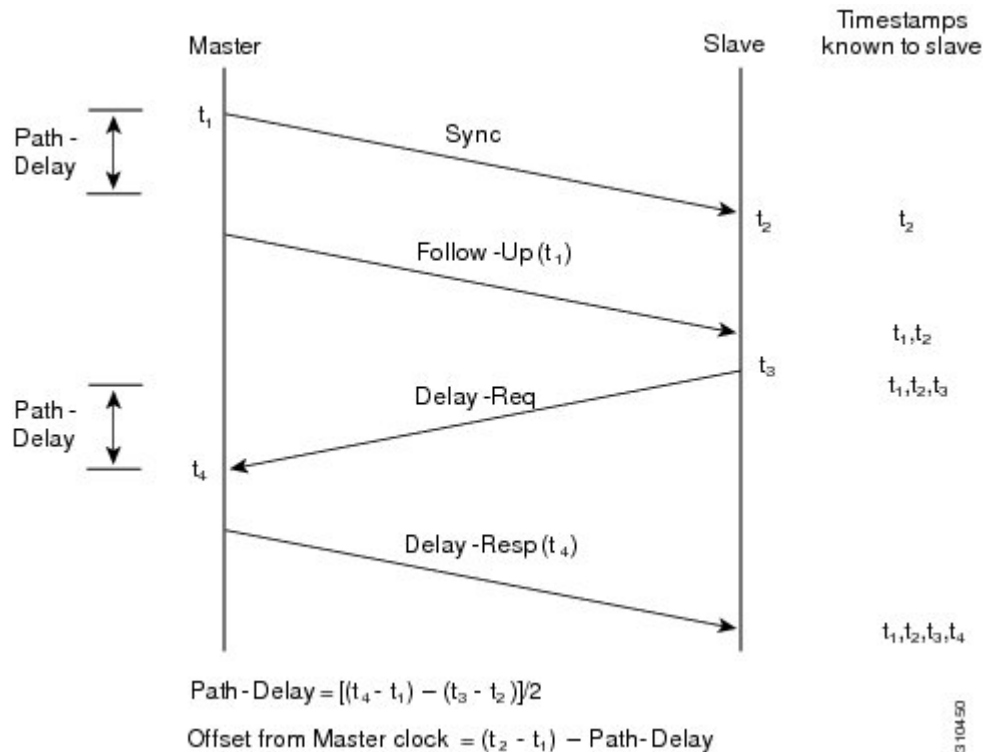
The ordinary and boundary clocks use **Sync**, **Delay_Req**, **Follow_Up**, **Delay_Resp** event messages to generate and communicate timing information.

These messages are sent in the following sequence:

1. The master sends a **Sync** message to the slave and notes the time, t_1 at which it was sent. For one-step **Sync** message carries the time when the message leaves the master and for two-step this time is sent in the subsequent **Follow-Up** event message.
2. The slave receives the **Sync** message and notes the time of reception, t_2 .
3. The master conveys to the slave the timestamp, t_1 by embedding the timestamp in a **Follow_Up** event message.
4. The slave sends a **Delay_Req** message to the master and notes the time, t_3 at which it was sent.
5. The master receives the **Delay_Req** message and notes the time of reception, t_4 .
6. The master conveys to the slave the timestamp, t_4 by embedding it in a **Delay_Resp** message.
7. After this sequence, the slave possesses all four timestamps. These timestamps can be used to compute the offset of the slave clock relative to the master, and the mean propagation time of messages between the two clocks.

The following figure describes the event messages in the PTP process that generate and communicate timing information.

Figure 4: PTP Process



ITU-T Telecom Profiles

Cisco NX-OS software supports ITU-T Telecom Profiles for PTP as defined in the ITU-T recommendation. A profile consists of PTP configuration options applicable only to a specific application.

Separate profiles can be defined to incorporate PTP in different scenarios based on the IEEE 1588-2008 standard. A telecom profile differs in several ways from the default behavior defined in the IEEE 1588-2008 standard and the key differences are mentioned in the subsequent sections.

The following sections describe the ITU-T Telecom Profiles that are supported for PTP:

Telecom Profile G.8275.1

Cisco's Telecom Profile G.8275.1 feature supports the ITU-T *G.8275.1 : Precision time protocol telecom profile for phase/time synchronization with full timing support from the network* standard. The G.8275.1 profile fulfills the time-of-day and phase synchronization requirements in telecom networks with all network devices participating in the PTP protocol. The G.8275.1 profile with SyncE provides better frequency stability for the time-of-day and phase synchronization.

Features of the G.8275.1 profile are:

- Synchronization Model: G.8275.1 profile adopts hop-by-hop synchronization model. Each network device in the path from master to slave synchronizes its local clock to upstream devices and provides synchronization to downstream devices.

- Clock Selection: G.8275.1 profile also defines an alternate BMCA that selects a clock for synchronization and port state for the local ports of all devices in the network is defined for the profile. The parameters defined as a part of the BMCA are:
 - Clock Class
 - Clock Accuracy
 - Offset Scaled Log Variance
 - Priority 2
 - Local Priority
 - Clock Identity
 - Steps Removed
 - Port Identity
- Port State Decision: The port states are selected based on the alternate BMCA.
- Alternate BMCA: It follows the alternate BMCA dataset comparison algorithm as defined in Rec. ITU-T G.8275.1/Y.1369.1 to select the GM for the node.
- Packet Rates: The nominal packet rate for Announce packets is 8 packets-per-second and 16 packets-per-second for Sync/Follow-Up and Delay-Request/Delay-Response packets.
- Transport Mechanism: G.8275.1 profile only supports Ethernet PTP transport mechanism.
- Mode: G.8275.1 profile supports transport of data packets only in multicast mode. The forwarding is done based on forwardable or non-forwardable multicast MAC address.
- Clock Type: G.8275.1 profile supports the following clock types:
 - Telecom Grandmaster (T-GM): Provides timing for other network devices and does not synchronize its local clock to other network devices.
 - Telecom Time Slave Clock (T-TSC): A slave clock synchronizes its local clock to another PTP clock, but does not provide PTP synchronization to any other network devices.
 - Telecom Boundary Clock (T-BC): Synchronizes its local clock to a T-GM or an upstream T-BC clock and provides timing information to downstream T-BC or T-TSC clocks.



Note Telecom Boundary Clock (T-BC) is the only clock type supported in Cisco NX-OS Release 9.3(5).

- Domain Numbers: The domain numbers that can be used in a G.8275.1 profile network ranges from 24 to 43. The default domain number is 24.

Telecom Profile G.8275.2

Cisco's Telecom Profile G.8275.2 feature supports the ITU-T G.8275.2 : *Precision time protocol telecom profile for phase/time synchronization with partial timing support from the network* standard. The G.8275.2 is a PTP profile for use in telecom networks where phase or time-of-day synchronization is required. It differs

from G.8275.1 in that it is not required that each device in the network participates in the PTP protocol. Also, G.8275.2 uses PTP over IPv4 and IPv6 in unicast mode.

The G.8275.2 profile is based on the partial timing support from the network. Hence, nodes using G.8275.2 are not required to be directly connected. The G.8275.2 profile is used in mobile cellular systems that require accurate synchronization of time and phase. For example, the fourth generation (4G) of mobile telecommunications technology.

With upcoming technologies such as LTE-TDD, LTE-A CoMP, LTE MBSFN and Location-based services, eNodeBs (base station devices) are required to be accurately synchronized in phase and time. Having GNSS systems at each node is not only expensive, but also introduces vulnerabilities. The G.8275.2 profile meets the synchronization requirements of these new technologies.

PTP Ports

A port can be configured to change its role dynamically. If no role is assigned to a port, it can dynamically assume a primary, passive, or subordinate role based on the BMCA.

In G.8275.2, PTP ports are not tied to any specific physical interfaces, but are tied to a loopback (virtual) interface. Traffic from a PTP port is routed through any physical interface based on the routing decision. For a dynamic port, only one clock source can be configured.

Alternate BPCA

The BPCA (Best Primary Clock Algorithm, which is also known as Best Master Clock Algorithm (BMCA [RFC 7273])) implementation in G.8275.2 is different from that in the default PTP profile. The G.8275.2 implementation specifies an alternate best primary clock algorithm (ABPCA), which is used by each device to select a clock to synchronize to, and to decide the port states of its local ports.

The following considerations apply to the G.8275.2 implementation of the BPCA:

- **PrimaryOnly**—A per port attribute, PrimaryOnly defines the state of the port. If this attribute is true, the port is never placed in the subordinate state.
- **Priority 1**—Priority 1 is always static in this profile and is set to 128. Priority 1 is not used in BPCA.
- **Priority 2**—Priority 2 is a configurable value and its range is from 0 to 255.
- **Local Priority**—Local priority is configured locally on clock ports to set the priority on nominated clocks. The default value is 128 and valid range is from 1 to 255.

Restrictions for Using the G.8275.2 Profile

- In the G.8275.2 profile, PTP is supported only in the none mode (default).
- A G.8275.2 PTP clock can have redundant clock sources configured (through multiple PTP ports). However, at any given time, a G.8275.2 PTP clock synchronizes to only one clock source, which is selected by BMCA.
- The G.8275.2 does not provide any recommendations for performance analysis and network limits for the clocks.

High Availability for PTP

Stateful restarts are not supported for PTP. After a reboot or a supervisor switchover, the running configuration is applied. For more information on high availability, see the [Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide](#).

Guidelines and limitations for PTP

**Note**

- For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.
- For PTP to function properly, you must use the latest SUP and line card FPGA versions. For information on upgrading FPGAs, access the [Release Notes landing page](#) and go to the FPGA/EPLD Upgrade Release Notes (NX-OS Mode Switches) section and locate the FPGA/EPLD Upgrade Release Notes for your software version. Refer to the Installation Guidelines topic.

General Guidelines

- PTP domain limits to a single domain per network.
- PTP can be enabled on the port-channel member ports.
- All management messages that are received from a slave port are forwarded to all PTP enabled ports. The management messages that are received from a slave port are not handled.
- To match PTP control packets using RACL, enable PIM on the L3 interface.
- When configuring PTP to Cisco Nexus 9000 Series switches, set the clock protocol to use PTP through the clock protocol ptp vdc 1 command.
- PTP is not available for all Cisco Nexus 9000 series and 3164Q hardware except for the 100G 9408PC line card and the 100G M4PC Generic Expansion Module (GEM).
- Before downgrading from Cisco NX-OS Release 9.3(1) to Release 7.0(3)I7, you must unconfigure PTP offload. PTP offloading is not supported for Cisco Nexus 9000 platform switches on 9636PQ, 9564PX, 9464PX, and 9536PQ line cards for Cisco NX-OS Release 7.0(3)I7.
- PTP-capable ports do not identify PTP packets and do not time-stamp or redirect those packets unless you enable PTP on those ports.
- The Cisco Nexus 9504 and 9508 platform switches with 9636C-R, 9636C-RX, or 9636Q-R line cards, the master ports can operate in one-step mode.
- In a topology where PTP is enabled, when a profile is set on a GrandMaster device, and the redundant GrandMasters are deployed in the network; to change the profile on the GrandMaster, you must first shut down the port that is configured on the GrandMaster to the switches, then change the profile, and then re-enable the port. For example, moving from an AES7 profile to an SMPTE profile or conversely.
- The QoS TCAM region Ingress SUP [ingress-sup] must be set to 768 or higher for PTP IPv6 transport to work.
- There is no CLI profile command for 8275.2. This will be added only when the APTS is supported. The functions for this release work only in default mode.

- Beginning with Cisco NX-OS Release 10.3(5)M, transmit and receive scope for PTP over IPv6 multicast configuration is supported.

Release Specific Guidelines

- Cisco NX-OS Release 10.4(3)F*
 - New global and interface commands are introduced to set the transmit and receive scope for PTP over IPv6 multicast configuration. If upgrading the devices to 10.4(3)F or higher, peer devices running NX-OS should also be upgraded to 10.4(3)F or higher for PTP over IPv6 multicast to work.



Note If any devices are not upgraded, then configure the workaround CLI **test system internal ptp ipv6-mcast-ipv4-dmac int-name** in the devices which are upgraded to 10.4(3)F.

- The following PTP profiles are supported on Cisco Nexus 9364C-H1 switch:
 - Default profile
 - Media profiles (AES67 and SMPTE 2059-2)
 - PTP telecom profile (only 8275-2)
- Cisco NX-OS Release 10.4(2)F*
 - 8275.2 PTP profile is supported on Cisco Nexus 9300-FX3, 9332D-H2R, 93400LD-H1, and 9408 platform switches. The guidelines for this profile are as follows:
 - Supports Unicast negotiation (simple unicast is not supported)
 - Default role for a dynamic unicast port is dynamic. The negotiation is triggered by state transitions. Only one PTP peer can be configured
 - Introduced a new command - **ptp peer** <ipv4>/<ipv6> - to configure a ptp peer when the role is dynamic.
 - For information about DME, see [Cisco Nexus NX-API References](#).
 - PTP Class C is now supported on N9K-C93400LD-H1 switches. The guidelines and limitations for this platform switches are as follows:
 - Only One-Step Mode is supported.
 - System can take a maximum of 30 minutes to achieve Class C level corrections
 - Both G.8275.1 and G.8275.2 Telecom profiles are supported, However, only Telecom Boundary Clock (T-BC) mode is supported.
 - Default PTP is supported on Cisco Nexus C93108TC-FX3 switch.
- Cisco NX-OS Release 10.4(1)F*
 - PTP is supported on the Cisco Nexus 9804 switches and the following guidelines and limitations apply:

- Only IPv4 transport is supported.
- Only media profile is supported.
- Only supported clock distribution mode is one step. Two step mode is not supported.
- Other PTP features such as unicast and unicast negotiation are not supported.
- Cisco Nexus X98900CD-A and X9836DM-A line cards with Cisco Nexus 9808 and 9804 switches
- PTP is not supported on Cisco Nexus C9348GC-FX3PH switch.
- PTP is supported on Cisco Nexus C9348GC-FX3 switch.

**Note**

Make sure that you increase COPP limit to have PTP state stable across PTP interfaces. For more information on configuration, see *CoPP Configuration Example* section in *Cisco Nexus 9000 Series NX-OS Security Configuration Guide*.

- ASIC-based time stamping on 1G PTP enabled ports is supported on N9K-C931806C-FX3 and N9K-C93108TC-FX3P platform switches. Thus, both 1G and 10G ports can run on ASIC mode now. However, all the 1G ports must be either in PHY mode (default) or in ASIC mode. A global command is introduced to enable this feature.
- PTP on media profile is supported on N9K-C9332D-H2R platform switch. A few guidelines and limitations for this platform switch are as follows:
 - Only multicast PTP with one-step is supported.
 - Unicast is not supported.
 - PTP is not supported on interface Ethernet 1/33 and Ethernet 1/34.
- PTP v1 packet forwarding feature is supported on Cisco Nexus N9K-C9408 and N9K-C9332D-H2R platform switches.
- The PTP Class C support for G.8275.1 Telecom profile feature is supported on Cisco Nexus 9408 and N9K-C9332D-H2R platform switches. A few guidelines and limitations for this platform switches are as follows:
 - The platform switches provide Class C support
 - Only One-Step Mode is supported
 - Ethernet transport is supported
 - System can take up to 30 minutes to achieve Class C level corrections
 - The G.8275.1 Telecom profile is supported, except the following:
 - Telecom Transparent Clock (T-TC)
 - Telecom Grand Master (T-GM)
- *Cisco NX-OS Release 10.3(5)M*

- New global and interface commands are introduced to set the transmit and receive scope for PTP over IPv6 multicast configuration. If upgrading the devices to 10.3(5)M or higher, peer devices running NX-OS should also be upgraded to 10.3(5)M or higher for PTP over IPv6 multicast to work.



Note If any devices are not upgraded, then configure the workaround CLI **test system internal ptp ipv6-mcast-ipv4-dmac int-name** in the devices which are upgraded to 10.3(5)M.

- *Cisco NX-OS Release 10.3(3)F*
 - 8275.2 Telecom profile is supported on Cisco Nexus 9300-FX3 platform switches. The following guidelines and limitations apply:
 - This feature allows ITU-T precision time protocol (PTP) profile for phase/time distribution with partial timing support from the network (unicast mode).
 - A command for G.8275.2 is added as the APTS is supported.
 - This feature also supports multiple unicast followers. Among the unicast followers, one is picked as an active unicast follower, while the rest configured unicast followers act as passive.
 - From this release onwards, dynamic role with unicast ports is supported.
 - For dynamic PTP unicast ports, the unicast-negotiation feature must be enabled.
- *Cisco NX-OS Release 10.3(2)F*
 - PTP GM functionality is supported on the Cisco Nexus N9K-C93180YC-FX3 platform switches.
 - PTP on media profile is supported on Cisco Nexus 9408 platform switches. A few guidelines and limitations for this platform switches are as follows:
 - Only multicast PTP with one-step is supported.
 - Other PTP features such as unicast and unicast negotiation are not supported.
- *Cisco NX-OS Release 10.3(1)F*
 - PTP Media Profile and One-Step Mode are supported on the Cisco Nexus 9808 platform switches. A few guidelines and limitations for this platform switches are as follows:
 - Only IPv4 transport is supported.
 - Only multicast PTP with one step is supported.
 - Other PTP features such as unicast and unicast negotiation are not supported.
- *Cisco NX-OS Release 10.2(3)F*
 - The PTP Support of up to 2000 Secondary Devices per Switch feature provides an option to support a maximum of 100 multicast secondary devices per port, with a system-wide support for a maximum of 2000 multicast secondary devices per switch. This feature is only supported on Cisco Nexus 9000-FX2 and 9000-FX3 platform switches.

- N9K-C9364D-GX2A - PTP is not supported on 1/65 and 1/66 ports
- N9K-C9348D-GX2A - PTP is not supported on 1/49 and 1/50 ports
- *Cisco NX-OS Release 10.2(2)F*
 - The PTPv1 and v2 co-existence feature is supported on the Cisco Nexus 9300-GX, 9300-GX2, and 9300-FX3 platform switches.
 - The PTP with jitter fix for 1G ports feature is supported on the Cisco N9K-C93108TC-FX3P platform switches.
 - The PTP unicast negotiation feature is also supported on Cisco Nexus 9300-EX, 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches.
 - The PTP IPv6 UDP transport feature is supported on the Cisco Nexus 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches.
- *Cisco NX-OS Release 10.2(1)F*
 - Unicast-negotiation is supported for IPv4 and IPv6 addresses with default profile on N9K-C93180YC-FX3S platform.
 - PTP IPv6 transport is supported on N9K-C93180YC-FX3S platform.
 - PTP is supported on the N9K-C9332D-GX2B platform switches. However, PTP is not supported on 1/33 and 1/34 ports.
- *Cisco NX-OS Release 10.1(2)F*
 - PTP is supported on the N9K-X9624D-R2 line cards.
 - PTP (IEEE 1588) is supported on the N9K-C9700-GX line card, as well as N9K-X97160YC-EX line card and N9K-C9700-FX line cards, used with N9K-C9504-FM-G and N9K-C9508-FM-G fabric modules.
 - PTP (IEEE 1588) is supported on the N9K-C9504-FM-G and N9K-C9508-FM-G platform switches.
 - Explicit carving of ingress-sup (size of the ingress supervisor TCAM region) to 768 is not required for the PTP profile 8275-1.
- *Cisco NX-OS Release 9.3(7)F*
 - PTP G.8275.1 Telecom profile is supported on N9K-C93180YC-FX3 platform switch.
- *Cisco NX-OS Release 9.3(5)F*
 - CLI behavior for PTP commands has changed as follows:
 - Most PTP commands do not return errors if the same command is applied again.
 - Most PTP commands do not validate the parameters that are entered as "no" commands. For example, if the currently configured command is "ptp sync interval -3", "no ptp sync interval -1" is accepted for negation.
 - PTP is supported on Cisco Nexus 9300-GX platform switches.
 - PTP G.8275.1 Telecom profile is supported on N9K-C93180YC-FX3S platform switch.

- PTP is supported on N9K-C93108TC-FX3P platform switch. However, syncE is not supported.
- *Cisco NX-OS Release 9.3(3)F*
 - PTP is supported on Cisco Nexus 93360YC-FX2 and 93216TC-FX2 switches.
- *Cisco NX-OS Release 9.3(2)F*
 - For Cisco Nexus 9636C-R, 9636Q-R, 9636C-RX, and 96136YC-R line cards, the following PTP configuration commands are defaults:
 - **ptp offload**
 - **ptp clock-operation one-step**

These default commands enable PTP to be distributed on the line cards and enable all PTP configured ports to run in one-step. We recommend using the default configuration for performance and scalability.



Note Changing the **ptp clock-operation** command to two step will disable the offload and PTP will then run centralized on the supervisor card.

- PTP offload is automatically enabled when one-step mode is configured.
- *Cisco NX-OS Release 9.2(3)F*
 - PTP is available for Cisco Nexus 9504-FM-R platform switches.
- *Cisco NX-OS Release 9.2(1)F*
 - Cisco Nexus 9732C-EX, 9736C-EX, and 97160YC-EX line cards support PTP offloading.

Telcom Profile Guidelines

- PTP Telecom Profile is supported only on the Cisco Nexus 93180YC-FX3S switch and N9K-C93180YC-FX3 switch.
- Only PTP class B is supported for 25G and above port speed.
- Only Telecom Boundary Clock (T-BC) is supported.
- Cisco's Telecom Profile G.8273.2 feature is compliant with the ITU-T G.8273.2 : *Timing characteristics of telecom boundary clocks and telecom time slave clocks* standard with the exception that 1PPS output is not aligned with PTP.



Note Time of Day and PTP GM are not supported in Cisco NX-OS Release 9.3(5).

Timestamp Tagging (TTAG) Support for Platform Switches

- Cisco Nexus 9200 platform switches: beginning with Cisco NX-OS Release 7.0(3)I6(1)

- Cisco Nexus 9364C: beginning with Cisco NX-OS Release 7.0(3)I7(2)
- Cisco Nexus 9332C: beginning with Cisco NX-OS Release 9.2(3)
- Cisco Nexus 9300-EX platform switches: beginning with Cisco NX-OS Release 7.0(3)I6(1)
- Cisco Nexus 9300-FX platform switches: beginning with Cisco NX-OS Release 7.0(3)I7(3)
- Cisco Nexus 9300-FX2 platform switches: beginning with Cisco NX-OS Release 9.3(3)
- Cisco Nexus 9300-FX3 and -GX platform switches: beginning with Cisco NX-OS Release 9.3(5)
- Cisco Nexus 9500 platform switches with -EX and -FX line cards

PTP Support

- PTP transport over User Datagram Protocol (UDP) is supported. PTP over Ethernet is supported only on Cisco Nexus 9300-FX3 platform switches.
- PTP supports the multicast communication. PTP also supports the unicast communication and the unicast mode is optional.
- PTP supports boundary clock mode.
- The PTP correction-range, PTP correction-range logging, and PTP mean-path-delay commands are supported on the Cisco Nexus 9508-R line cards.
- Cisco Nexus 9000 series switches support mixed non-negotiated mode of operation on master PTP ports. That means when a slave client sends unicast delay request PTP packet, the Cisco Nexus 9000 responds with a unicast delay response packet. And, if the slave client sends multicast delay request PTP packet, the Cisco Nexus 9000 responds with a multicast delay response packet. For mixed non-negotiated mode to work, the source IP address used in the `ptp source IP address` configuration on the BC device must also be configured on any physical or logical interface of the BC device. The recommended best practice is to use the loopback interface of the device.
- Cisco Nexus 93108TC-EX and 93180YC-EX switches support PTP in mixed mode and unicast mode. The Cisco Nexus 9396 switch supports PTP mixed mode.
- PTP is supported with sync interval -3 only on Cisco Nexus 9508-R family line cards. Higher sync intervals are not supported.
- PTP unicast is supported only on the default vrf.
- PTP supports mixed mode for delivering PTP messages, which is detected automatically by a Cisco Nexus device based on the type of delay request message that is received from a connected client and no configuration is required.
- PTP supports unicast communication on Cisco Nexus 9508 switches with 9636C-R, 9636C-RX, and 9636Q-R line cards.
- PTP one-step mode is supported only in PTP offload mode for the Cisco Nexus 9504 and 9508 platform switches with 9636C-R, 9636C-RX, 9624D-R2, or 9636Q-R line cards.
- Each port can be individually configured with any of the supported PTP profiles. Different PTP profiles can coexist on an interface. Combination of the default of 1588 and SMPTE-2059-2 or AES67 profiles is supported.
- The platform switches are supported only on Class B and do not meet Class C support.

- One-step PTP is only supported on Cisco Nexus 9000-R series platform switches.

PTP Not Supported

- PTPv1 forwarding and feature VMCT are not supported if enabled at the same time.
- 1 Pulse per Second (1PPS) output is enabled by default. UTC/SMB port is in output mode. Note that 1PPS output is not supported.
- End-to-end transparent clock and peer-to-peer transparent clock modes are not supported.
- For Cisco Nexus 31108PC-V and 31108TC-V switches, PTP is not supported on ports running at 100G speed.
- PTP is not supported for stateful high availability.
- PTP is not supported for management interfaces.
- PTP unicast is not supported in offload mode.
- PTP configuration with UC and MC on either side is not supported on Cisco Nexus 9508 switches with 9636C-R, 9636C-RX, and 9636Q-R line cards.
- PTP is not supported on FEX interfaces.
- A combination of SMPTE-2059-2 and AES67 profiles is not supported on the same interface.
- PTP is not supported on N9K-C92348GC-X.

Default PTP Parameters

The following table lists the default settings for PTP parameters.

Table 3: Default PTP Parameters

Parameters	Default
PTP	Disabled
PTP version	2
PTP domain	0
PTP priority 1 value when advertising the clock	255
PTP priority 2 value when advertising the clock	255
PTP announce interval	1 log second
PTP announce timeout	3 announce intervals
PTP delay-request interval	• 0 log seconds • -1 log seconds for Cisco Nexus 3232C, 3264Q, and 9500 platform switches

Parameters	Default
PTP sync interval	• -2 log seconds • -3 log seconds for Cisco Nexus 3232C, 3264Q, and 9500 platform switches
PTP VLAN	gPTP supports only default vlan 1, and no other user configured VLANs.

Configure PTP

Configure PTP Globally

You can enable or disable PTP globally on a device. You can also configure various PTP clock parameters to help determine which clock in the network has the highest priority to be selected as the grandmaster.



Note For PTP to function properly, you must use the latest SUP and LC FPGA versions. For information on upgrading FPGAs, access the [Release Notes landing page](#) and go to the FPGA/EPLD Upgrade Release Notes (NX-OS Mode Switches) section and locate the FPGA/EPLD Upgrade Release Notes for your software version. Refer to the Installation Guidelines topic.



Note You must always set the clock protocol PTP vdc1 for the local clock to be updated by the PTP protocol, irrespective of the one-step or the two-step mode. You can verify the configuration using the **show running-config clock_manager** command.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] feature ptp Example: <pre>switch(config)# feature ptp</pre>	Enables or disables PTP on the device. Note Enabling PTP on the switch does not enable PTP on each interface. Make sure that only one of these features is configured: dot1x (feature dot1x) or NV overlay (feature nv overlay). A dynamic ifacl

	Command or Action	Purpose
		label is reserved when these features are configured. However, only two dynamic ifacl label bits are available. If both of these features are already configured, a dynamic ifacl label will not be available for PTP, and the feature cannot be enabled.
Step 3	[no] ptp device-type [generalized-ptp boundary-clock ordinary-clock-grandmaster] Example: <pre>switch(config)# ptp device-type generalized-ptp</pre>	Configures the device type as gPTP or boundary clock or ordinary clock grandmaster. Note The generalized-ptp option is available only for Cisco Nexus 9508 switches with an -R series line card. Note Beginning with Cisco NX-OS Release 10.3(2)F, the ordinary clock grandmaster is available only for Cisco Nexus N9K-C93180YC-FX3 platform switches. For more information, refer to Configure PTP GM, on page 79.
Step 4	[no] ptp source {<ipv4 address> <ipv6 address>} [vrf <vrf-name>] Example: <pre>switch(config)# ptp source 10.10.10.1</pre>	Configures the source IPv4/IPv6 address for all the PTP packets in the multicast PTP mode. If the IP address is in non-default VRF, use the vrf parameter to configure the VRF. Corresponding source address (IPv4/IPv6) is needed before enabling PTP IPv4/IPv6 transport on an interface. Note IPv6 source is supported on Cisco Nexus 93180TC-FX3S switch starting with 10.2(1)F release. Beginning with Cisco NX-OS Release 10.2(2)F, this option is also available on Cisco Nexus 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches.
Step 5	(Optional) [no] ptp domain <i>number</i> Example: <pre>switch(config)# ptp domain 1</pre>	Configures the domain number to use for this clock. PTP domains allow you to use multiple independent PTP clocking subdomains on a single network. The range for the <i>number</i> is from 0 to 127.
Step 6	(Optional) [no] ptp offload Example:	Increases the number of PTP sessions by offloading some timers to the line card.

	Command or Action	Purpose
	<pre>switch(config)# ptp offload</pre>	This step is required for one-step mode and optional for two-step mode. Note Make sure that neither of these features are already configured: dot1x (feature dot1x) and NV overlay (feature nv overlay). A dynamic ifacl label is reserved when these features are configured. However, only two dynamic ifacl label bits are available. If either of these features are already configured, a dynamic ifacl label will not be available for PTP offload, and the feature cannot be enabled. Note that PTP (feature ptp) consumes one ifacl label. Note Beginning with Cisco NX-OS Release 9.3(3), Cisco Nexus 9504 and 9508 platform switches with 9636C-R, 9636C-RX, or 9636Q-R line cards support offload only with one-step clock operation. PTP offload is automatically enabled or disabled when the one-step clock operation is enabled or disabled.
Step 7	(Optional) [no] ptp clock-operation one-step Example: <pre>switch(config)# ptp clock-operation one-step</pre>	Configures the PTP clock operation to the one-step mode. In this case, the timestamp message is sent as a part of the sync message. A followup message is not sent in this mode.
Step 8	(Optional) [no] ptp priority1 value Example: <pre>switch(config)# ptp priority1 1</pre>	Configures the priority1 value to use when advertising this clock. This value overrides the default criteria (clock quality, clock class, and so on) for best master clock selection. Lower values take precedence. The range for the <i>value</i> is from 0 to 255. Note For the switch to synchronize with an external Grand Master clock, the local switch PTP priority value must be configured higher than that of external Grand Master Clock priority.
Step 9	(Optional) [no] ptp priority2 value Example: <pre>switch(config)# ptp priority2 1</pre>	Configures the priority2 value to use when advertising this clock. This value is used to decide between two devices that are otherwise equally matched in the default criteria. For example, you can use the priority2 value to give a specific switch priority over other identical switches.

	Command or Action	Purpose
		The range for the <i>value</i> is from 0 to 255. Note For the switch to synchronize with an external Grand Master clock, the local switch PTP priority value must be configured higher than that of external Grand Master Clock priority.
Step 10	[no] ptp management Example: <pre>switch(config)# ptp management switch(config-ptp-profile)#</pre>	Configures support for PTP management packets. This command is enabled by default. no: Disables support for management packets.
Step 11	(Optional) [no] ptp delay tolerance { mean-path reverse-path } variation Example: <pre>switch(config)# ptp delay tolerance mean-path 50.5 switch(config)#</pre>	Configures the PTP delay mean path/reverse path tolerance variation. mean-path: Ignore spikes in Mean Path Delay (MPD) as calculated by the PTP BMC algorithm. reverse-path: Ignore spikes in (t4-t3) as calculated by the PTP BMC algorithm. variation: Percentage that defines the tolerance for spikes. Use numeric values with a single decimal. Range is from 1.0 through 100.0. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).
Step 12	(Optional) ptp forward-version1 Example: <pre>switch(config)# ptp forward-version1 switch(config)#</pre>	Configures the switch to forward all PTPv1 packets based on the forwarding rule. Note If you do not enable this command, all PTPv1 packets are passed on to the CPU and ultimately dropped. This command is supported beginning with Cisco NX-OS Release 9.3(6).
Step 13	(Optional) ptp unicast-negotiation	This configuration is introduced in Cisco Nexus NX-OS Release 10.2(1)F and is supported on 93180YC-FX3S. From Cisco NX-OS Release 10.2(2)F onwards, this configuration is supported on Cisco Nexus 9300-EX, 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches. When enabled, all PTP unicast sessions will transition to negotiated mode.

	Command or Action	Purpose
		For more information, refer to the PTP Unicast-Negotiation section.
Step 14	(Optional) ptp ipv6 multicast scope tx <i><scope></i> Example: <pre>switch(config)# ptp ipv6 multicast scope tx 0x2</pre>	Sets the transmit scope for PTP over IPv6 multicast configuration. Note • Transmit scope is a single value. This determines the scope of IPv6 multicast destination address (FF0:X') used by PTP packets transmitted by the switch. • The default value is 1 (0x1). • When interface level configuration is applied, it takes precedence over the corresponding global configuration.
Step 15	(Optional) ptp ipv6 multicast scope rx <i><scope></i> Example: <pre>switch(config)# ptp ipv6 multicast scope rx 0x1-0x5,0x8,0xA-0xE</pre>	Sets the receive scope for PTP over IPv6 multicast configuration. Note • Receive scope can be a single value or multiple values or a range of values. This determines the acceptable scope (FF0:X') for received IPv6 multicast PTP packets. • The default value is 1 through 0xE (0x1-0xE). • When interface level configuration is applied, it takes precedence over the corresponding global configuration.
Step 16	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure PTP GM

Beginning with Cisco NX-OS Release 10.3(2)F, the following are the newly added CLIs to support PTP GM functionality:

Procedure

	Command or Action	Purpose
Step 1	[no] ptp device-type ordinary-clock-grandmaster Example: <pre>switch(config)# ptp device-type ordinary-clock-grandmaster</pre>	Configures the device type as ordinary clock grandmaster.
Step 2	(Optional) ptp utc-offset <leap-seconds> [next-leapevent <date> <time> new-offset <new-leap-seconds>] Example: <pre>switch(config)# ptp utc-offset 37 next-leap-event 2022-04-30 23:59:59 new-offset 38</pre>	Configures the utc-offset value for the PTP GM functionality. Default value is 37 seconds. • utc-offset: UTC offset value to be added to GM clock. <i>leap-seconds</i>: Leap seconds value, range is 0–125 seconds. The default value is 37 seconds. next-leapevent: Optional keyword to configure time at which leap seconds change. <i>date</i>: Date at which leap seconds value changes in YYYY-MM-DD format. <i>time</i>: Time at which leap seconds value will change in HH:MM:SS format new-offset: New UTC offset value to be used after above configured utc-offset value. <i>new-leap-seconds</i>: New leap seconds value, range is 0–125 seconds. It can only be ± 1 of the configured utc-offset value.
Step 3	clock protocol gnss Example: <pre>switch(config)# clock protocol gnss</pre>	Enables the GNSS Time of Day (ToD) sync to system clock on the device.
Step 4	For external GNSS configuration, refer to <i>Configuring GPS Settings for the Grandmaster Clock</i> section. For internal GNSS configuration, refer to Enabling the GNSS Receiver, on page 126 section.	

Configure PTP on an Interface

After you globally enable PTP, it is not enabled on all supported interfaces by default. You must enable PTP interfaces individually.

Before you begin

Make sure that you have globally enabled PTP on the switch and configured the source IP address for PTP communication.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface ethernet slot/port Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Specifies the interface on which you are enabling PTP and enters the interface configuration mode.
Step 3	[no] ptp Example: <pre>switch(config-if)# ptp</pre>	Enables or disables PTP on an interface.
Step 4	(Optional) ptp transport {ethernet ipv4 ipv6 } Example: <pre>switch(config-if)# ptp transport ipv4 switch(config-if)# switch(config-if)# ptp transport ipv6 switch(config-if)#</pre>	Specifies the transport mechanism that is used to send PTP packets. ethernet: PTP packets are carried only in Eth frame (Eth/ptp). This option is only available for PTP Telecom Profile on the Cisco Nexus 93180YC-FX3S switch. ipv4: PTP packets are carried over IPv4. This is the default setting. ipv6: PTP packets are carried over IPv6. This option is available on Cisco Nexus 93180YC-FX3S switch starting with 10.2(1)F release. Beginning with Cisco NX-OS Release 10.2(2)F, this option is also available on Cisco Nexus 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).
Step 5	(Optional) ptp transmission {multicast unicast [negotiation-schema <schema-name>]} Example:	Configures the PTP transmission method that is used by the interface. multicast: PTP uses multicast destination IP address 224.0.1.129 as per IEEE 1588 standards for communication between devices. This is the default setting.

	Command or Action	Purpose
	<pre>switch(config-if)# ptp transmission multicast switch(config-if)#</pre>	unicast: PTP messages are unicast to a particular PTP peer node. negotiation schema <schema-name>: This option can be used when unicast-negotiation is enabled globally and can be used set the negotiation schema to be used on the interface. This option is available on the Cisco Nexus 93180YC-FX3S switch beginning with Cisco NX-OS Release 10.2(1)F. Beginning with Cisco NX-OS Release 10.2(2)F, this option is also available on Cisco Nexus 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).
Step 6	(Optional) ptp role { dynamic master slave } Example: <pre>switch(config-if)# ptp role dynamic switch(config-if)#</pre>	Configures the PTP role of the interface. dynamic: The best master clock algorithm (BMCA) assigns the role. This is the default setting for the default PTP profile and the only allowed setting for the G.8275.1 PTP profile. Beginning with Cisco NX-OS Release 10.3(3)F, if the PTP transmission is unicast, and the PTP role is set to dynamic, enable PTP unicast-negotiation. master: The master clock is assigned as the PTP role of the interface. slave: The slave clock is assigned as the PTP role of the interface. Beginning with Cisco NX-OS Release 10.3(3)F, PTP role slave can be configured on multiple unicast ports. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).
Step 7	(Optional) [no] ptp master { <ipv4-addr> <ipv6-addr> } { negotiation-schema <schema-name> } Example: <pre>switch(config-if)# ptp master 10.10.10.1 switch(config-if)#</pre>	(Optional) Sets the IP address of the master clock when the PTP role of the interface is set to "slave". negotiation-schema : This can be used to set specific negotiation schema for the master when unicast-negotiation is enabled globally. This option is available on the Cisco Nexus

	Command or Action	Purpose
		93180YC-FX3S switch beginning with Cisco NX-OS Release 10.2(1)F. Beginning with Cisco NX-OS Release 10.2(2)F, this option is also available on Cisco Nexus 9300-EX, 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches. Note This command configures the unicast master and is used when transmission is set to unicast. This command is supported beginning with Cisco NX-OS Release 9.3(5). IPv6 is supported on Cisco Nexus 93180YC-FX3S beginning with Cisco NX-OS Release 10.2(1)F. Beginning with Cisco NX-OS Release 10.2(2)F, IPv6 is also supported on Cisco Nexus 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches.
Step 8	(Optional) [no] ptp slave { <ipv4-addr> <ipv6-addr> } Example: <pre>switch(config-if)# ptp slave 10.10.10.2 switch(config-if)#</pre>	(Optional) Sets the IP address of the slave clock when the PTP role of the interface is set to "master". Note This command configures the unicast slave and is used when transmission is set to unicast. This command is supported beginning with Cisco NX-OS Release 9.3(5). IPv6 is supported on Cisco Nexus 93180YC-FX3S beginning with Cisco NX-OS Release 10.2(1)F. Beginning with Cisco NX-OS Release 10.2(2)F, IPv6 is supported on Cisco Nexus 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches.
Step 9	(Optional) ptp peer ipv4-address [vrf vrf-name] Example: <pre>switch(config-if)# ptp peer 10.10.10.2 switch(config-if)#</pre>	Sets the IP address of the peer clock when the PTP role of the interface is set to dynamic. Note • You can configure only one peer. • This command configures the unicast peer and is used when transmission is set to unicast and role is set to dynamic. • This command is introduced in Cisco NX-OS Release 10.3(3)F.

	Command or Action	Purpose
Step 10	ptp multicast master-only Example: <pre>switch(config)# ptp multicast master-only switch(config)#</pre>	Configures the master clock that is assigned as the PTP role of the interface. Note This command is deprecated in Cisco NX-OS Release 9.3(5) and is not supported in future releases. Please use commands in Steps 4-8 as applicable.
Step 11	(Optional) ptp ucast-source {<ipv4-addr> <ipv6-addr>} [vrf <vrf-name>] Example: <pre>switch(config)# ptp ucast-source 10.1.1.40</pre>	(Optional) Configures the source IP address for unicast messages. <i>ipv4-address</i>: The IPv4 address of the unicast source. This is used when transport is set to IPv4. <i>ipv6-address</i>: The IPv6 address of the unicast source. This is used when transport is set to IPv6. vrf <i>vrf-name</i>: The name of the VRF used for hello messages. Note IPv6 is supported on Cisco Nexus 93180YC-FX3S beginning with Cisco NX-OS Release 10.2(1)F. Beginning with Cisco NX-OS Release 10.2(2)F, IPv6 is also supported on Cisco Nexus 9300-FX, 9300-FX2, 9300-GX, and 9300-GX2 platform switches.
Step 12	(Optional) [no] ptp announce {interval <i>log-seconds</i> timeout <i>count</i>} Example: <pre>switch(config-if)# ptp announce interval 3</pre>	Configures the interval between PTP announce messages on an interface or the number of PTP intervals before a timeout occurs on an interface. The range for the PTP announcement interval is from 0 to 4 log seconds, and the range for the interval timeout is from 2 to 4 intervals.
Step 13	(Optional) [no] ptp delay-request minimum interval <i>log-seconds</i> Example: <pre>switch(config-if)# ptp delay-request minimum interval -1</pre>	Configures the minimum interval allowed between PTP delay messages when the port is in the master state. The range is from log(-1) to log(6) seconds, where log(-1) = 2 frames every second.
Step 14	(Optional) [no] ptp delay-request minimum interval [aes67-2015 smppte-2059-2] <i>log-seconds</i> Example:	Configures the minimum interval allowed between PTP delay messages when the port is in the master state.

	Command or Action	Purpose												
	<pre>switch(config-if)# ptp delay-request minimum interval aes67-2015-1</pre>	Table 4: PTP Delay-Request Minimum Interval Range and Default Values <table> <tr> <th>Option</th><th>Range</th><th>Default Value</th></tr> <tr> <td>aes67-2015</td><td>–4 to 5 log seconds</td><td>0 log seconds</td></tr> <tr> <td>smppte-2059-2</td><td>–4 to 5 log seconds</td><td>0 log seconds</td></tr> <tr> <td>Without the aes67-2015 or smppte-2059-2 option</td><td>–1 to 6 log seconds (where –1 = 2 frames every second)</td><td>0 log seconds</td></tr> </table>	Option	Range	Default Value	aes67-2015	–4 to 5 log seconds	0 log seconds	smppte-2059-2	–4 to 5 log seconds	0 log seconds	Without the aes67-2015 or smppte-2059-2 option	–1 to 6 log seconds (where –1 = 2 frames every second)	0 log seconds
Option	Range	Default Value												
aes67-2015	–4 to 5 log seconds	0 log seconds												
smppte-2059-2	–4 to 5 log seconds	0 log seconds												
Without the aes67-2015 or smppte-2059-2 option	–1 to 6 log seconds (where –1 = 2 frames every second)	0 log seconds												
Step 15	(Optional) [no] ptp sync interval log-seconds Example: <pre>switch(config-if)# ptp sync interval 1</pre>	Configures the interval between PTP synchronization messages on an interface. The range is from log(–3) to log(1) seconds. For the media-related profile information, see the Cisco NX-OS IP Fabric for Media Solution Guide when configuring PTP for media.												
Step 16	(Optional) [no] ptp sync interval [aes67-2015 smppte-2059-2] log-seconds Example: <pre>switch(config-if)# ptp sync interval aes67 1</pre>	Configures the interval between PTP synchronization messages on an interface. Table 5: PTP Synchronization Interval Range and Default Values <table> <tr> <th>Option</th><th>Range</th><th>Recommended Value</th></tr> <tr> <td>aes67-2015</td><td>–7 to 0 log seconds</td><td>–3 log seconds</td></tr> <tr> <td>smppte-2059-2</td><td>–7 to –0 log seconds</td><td>–3 log seconds</td></tr> </table>	Option	Range	Recommended Value	aes67-2015	–7 to 0 log seconds	–3 log seconds	smppte-2059-2	–7 to –0 log seconds	–3 log seconds			
Option	Range	Recommended Value												
aes67-2015	–7 to 0 log seconds	–3 log seconds												
smppte-2059-2	–7 to –0 log seconds	–3 log seconds												
Step 17	(Optional) [no] ptp vlan vlan-id Example: <pre>switch(config-if)# ptp vlan 1</pre>	Specifies the VLAN for the interface where PTP is being enabled. You can only enable PTP on one VLAN on an interface. The range is from 1 to 4094.												
Step 18	(Optional) ptp destination-mac non-forwardable rx-no-match accept Example:	Accepts and responds to nonforwardable destination MAC address packets. These destination MAC addresses are used in the PTP messages that are exchanged between the GM												

	Command or Action	Purpose
	<pre>switch(config-if)# ptp destination-mac non-forwardable rx-no-match accept switch(config-if)#</pre>	clock, PTP-master clock, and PTP-slave clocks. This command is supported beginning with Cisco NX-OS Release 9.3(5) and only on the Cisco Nexus 93180YC-FX3S switch.
Step 19	(Optional) show ptp brief Example: <pre>switch(config-if)# show ptp brief</pre>	Displays the PTP status.
Step 20	(Optional) show ptp port interface interface slot/port Example: <pre>switch(config-if)# show ptp port interface ethernet 2/1</pre>	Displays the status of the PTP port.
Step 21	(Optional) ptp ipv6 multicast scope tx <scope> Example: <pre>switch(config)# ptp ipv6 multicast scope tx 0x2</pre>	Sets the transmit scope for PTP over IPv6 multicast configuration. Note • Transmit scope is a single value. This determines the scope of IPv6 multicast destination address (FF0'X') used by PTP packets transmitted by the switch. • The default value is 1 (0x1). • When interface level configuration is applied, it takes precedence over the corresponding global configuration.
Step 22	(Optional) ptp ipv6 multicast scope rx <scope> Example: <pre>switch(config)# ptp ipv6 multicast scope rx 0x1-0x5,0x8,0xA-0xE</pre>	Sets the receive scope for PTP over IPv6 multicast configuration. Note • Receive scope can be a single value or multiple values or a range of values. This determines the acceptable scope (FF0'X') for received IPv6 multicast PTP packets. • The default value is 1 through 0xE (0x1-0xE). • When interface level configuration is applied, it takes precedence over the corresponding global configuration.

	Command or Action	Purpose
Step 23	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure PTP in Unicast Mode

Configure Unicast Mode for IPv4 or IPv6

Traditional PTP messages are delivered to the nodes that are capable of receiving PTP multicast messages. (For example, **announce**, **sync**, **delay_req**, **delay_resp** and **follow_up**). In Unicast mode, all PTP messages are delivered only to a particular PTP node. Multicast address is not used. In unicast mode, you can configure master/slave role and assign corresponding peer slave/master IP addresses.

Up to 8 master IPs can be configured for a slave unicast port and 64 slave IPs can be configured for a master port with a maximum 256 slave IP total for all ports. The following commands are used to configure the unicast slave IPs and unicast master IPs. Unicast packets are only sent to and received from these IPs. Packets received from other IPs are ignored.

For Cisco NX-OS Release 10.2(1)F and later:

```
IPv4 config
interface Ethernet1/34
 ptp
 ptp transport ipv4
 ptp transmission unicast
 ptp role master
 ptp slave 10.10.10.2
 ptp ucast-source 10.10.10.1

interface Ethernet1/35
 ptp
 ptp transport ipv4
 ptp transmission unicast
 ptp role slave
 ptp master 10.10.10.1
 ptp ucast-source 10.10.10.2

IPv6 config
interface Ethernet1/34
 ptp
 ptp transport ipv6
 ptp transmission unicast
 ptp role master
 ptp slave 2012:a1:0:0:0:0:0:2
 ptp ucast-source 2012:a1:0:0:0:0:0:1

interface Ethernet1/35
 ptp
 ptp transport ipv6
 ptp transmission unicast
 ptp role slave
 ptp master 2012:a1:0:0:0:0:0:1
 ptp ucast-source 2012:a1:0:0:0:0:0:2
```

For Cisco NX-OS Release 9.3(5) and later:

```
switch(config-if)# ptp
switch(config-if)# ptp transmission unicast
switch(config-if)# ptp role master
switch(config-if)# ptp slave 10.10.10.2
```

```
switch(config-if)# ptp
switch(config-if)# ptp transmission unicast
switch(config-if)# ptp role slave
switch(config-if)# ptp master 10.10.10.1
```

For Cisco NX-OS Release 9.3(4) and earlier:

```
switch(config-if)# ptp transport ipv4 ucast master
switch(config-if-ptp-master)# slave ipv4 10.10.10.2
```

```
switch(config-if)# ptp transport ipv4 ucast slave
switch(config-if-ptp-slave)# master ipv4 10.10.10.1
```

Assign Dynamic Role

Complete the following steps to assign a dynamic role:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface ethernet <i>slot/port</i> Example: switch(config)# interface ethernet 2/1 switch(config-if)#	Specifies the interface on which you are enabling PTP and enters the interface configuration mode.
Step 3	[no] ptp Example: switch(config-if)# ptp switch(config-if)#	Enables or disables PTP on an interface. Note This command is required prior to applying below unicast configuration commands on the interface.
Step 4	ptp transmission unicast Example: switch(config-if)# ptp transmission unicast switch(config-if)#	Configures the PTP transmission method that is used by the interface.
Step 5	ptp role dynamic Example: switch(config-if)# ptp role dynamic switch(config-if)#	Configures the PTP role of the interface. dynamic: The dynamic clock is assigned as the PTP role of the interface.

	Command or Action	Purpose
Step 6	ptp peer <i>ipv4-address</i> Example: <pre>switch(config-if)# ptp peer 10.10.10.1 switch(config-if)#</pre>	Sets the IP address of the peer clock when the PTP role of the interface is set to dynamic.

Assign Master Role

Complete the following steps to assign a master role:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface ethernet <i>slot/port</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Specifies the interface on which you are enabling PTP and enters the interface configuration mode. Note After configuring this command, for Cisco NX-OS Release 9.3(5) and later, skip to step 5. For Cisco NX-OS Release 9.3(4) and earlier, continue with step 3.
Step 3	[no] ptp transport ipv4 ucast master Example: <pre>switch(config-if)# ptp transport ipv4 ucast master switch(config-if-ptp-master)#</pre>	Enables PTP master on a particular port (Layer 3 interface). In the master sub-mode, you can enter the slave IPv4 addresses.
Step 4	slave ipv4 <IP_address> Example: <pre>switch-1(config)# interface ethernet 1/1 switch-1(config-if)# ptp transport ipv4 ucast master switch-1(config-if-ptp-master)# slave ipv4 1.2.3.1 switch-1(config-if-ptp-master)# slave ipv4 1.2.3.2 switch-1(config-if-ptp-master)# slave ipv4 1.2.3.3 switch-1(config-if-ptp-master)# slave ipv4 1.2.3.4 switch-1(config-if-ptp-master)#</pre>	Enters the slave IPv4 addresses. Maximum of 64 IP addresses are allowed per master, but this number varies and it depends on the sync interval configuration. The master sends announce, sync, follow-up, and delay_resp only to these slave addresses. You must make sure that the slave IP is reachable. Note For Cisco NX-OS Release 9.3(4) and earlier, this concludes the procedure.
Step 5	[no] ptp	Enables or disables PTP on an interface.

	Command or Action	Purpose
	Example: <pre>switch(config-if)# ptp switch(config-if)#</pre>	Note Starting with 9.3(5), this command is required prior to applying below unicast configuration commands on the interface.
Step 6	ptp transmission unicast Example: <pre>switch(config-if)# ptp transmission unicast switch(config-if)#</pre>	Configures the PTP transmission method that is used by the interface. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).
Step 7	ptp role master Example: <pre>switch(config-if)# ptp role master switch(config-if)#</pre>	Configures the PTP role of the interface. master: The master clock is assigned as the PTP role of the interface. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).
Step 8	ptp slave ipv4-address Example: <pre>switch(config-if)# ptp slave 10.10.10.2 switch(config-if)#</pre>	Sets the IP address of the slave clock when the PTP role of the interface is set to "master". Note This command is supported beginning with Cisco NX-OS Release 9.3(5).

Assign Slave Role

Complete the following steps to assign a slave role:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface ethernet slot/port Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Specifies the interface on which you are enabling PTP and enters the interface configuration mode. Note After configuring this command, for Cisco NX-OS Release 9.3(5) and later, skip to step 5. For Cisco NX-OS Release 9.3(4) and earlier, continue with step 3.

	Command or Action	Purpose
Step 3	[no] ptp transport ipv4 ucast slave Example: <pre>switch(config-if)# ptp transport ipv4 ucast slave switch(config-if-ptp-slave)#</pre>	Enables PTP slave on a particular port (Layer 3 interface). In the slave sub-mode, you can enter the master IPv4 addresses.
Step 4	master ipv4 <IP_address> Example: <pre>switch-1(config)# interface ethernet 1/1 switch-1(config-if)# ptp transport ipv4 ucast slave switch-1(config-if-ptp-slave)# master ipv4 4.4.4.1 switch-1(config-if-ptp-slave)# master ipv4 4.4.4.2 switch-1(config-if-ptp-slave)# master ipv4 4.4.4.3</pre>	Enters the master IPv4 addresses. Note For Cisco NX-OS Release 9.3(4) and earlier, this concludes the procedure.
Step 5	[no] ptp Example: <pre>switch(config-if)# ptp switch(config-if)#</pre>	Enables or disables PTP on an interface. Note Starting with 9.3(5), this command is required prior to applying below unicast configuration commands on the interface
Step 6	ptp transmission unicast Example: <pre>switch(config-if)# ptp transmission unicast switch(config-if)#</pre>	Configures the PTP transmission method that is used by the interface. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).
Step 7	ptp role slave Example: <pre>switch(config-if)# ptp role slave switch(config-if)#</pre>	Configures the PTP role of the interface. slave: The slave clock is assigned as the PTP role of the interface. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).
Step 8	ptp master ipv4-address Example: <pre>switch(config-if)# ptp master 10.10.10.1 switch(config-if)#</pre>	Sets the IP address of the master clock when the PTP role of the interface is set to "slave". Note This command is supported beginning with Cisco NX-OS Release 9.3(5).

Configure Unicast Source Address



Note For all releases up to, and including Cisco NX-OS Release 9.3(4), if the PTP configuration on the interface is changed from unicast to multicast or unicast slave to unicast master, you must reconfigure the unicast source address.

Beginning with Cisco NX-OS Release 9.3(5), if the PTP configuration on the interface is changed from unicast to multicast or unicast slave to unicast master, you do not need to reconfigure the unicast source address.

Complete the following steps to configure unicast source address:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface ethernet <i>slot/port</i> Example: <pre>switch(config)# interface ethernet 2/1 switch(config-if)#</pre>	Specifies the interface on which you are enabling PTP and enters the interface configuration mode.
Step 3	[no] ptp ucast-source {<ipv4-addr> <ipv6-addr>} [vrf <vrf-name>] Example: <pre>switch(config-if)# ptp ucast-source 10.10.10.20 switch(config-if)#</pre>	Configure PTP source address per interface level. This IP address is used only for unicast PTP messages. The PTP unicast source IP address must be reachable. <i>ipv4-address</i>: The IPv4 address of the unicast source. This is used when transport is set to IPv4. <i>ipv6-address</i>: The IPv6 address of the unicast source. This is used when transport is set to IPv6. <i>vrfvrf-name</i>: If the IP address is in non-default VRF, use the vrf parameter to configure the VRF.

Configure PTP Telecom Profile

Configure Global PTP Telecom Profile

This procedure describes the steps involved to configure PTP telecom profile including the clock and its settings to be consistent with ITU-T Telecom Profiles for Frequency.

Before you begin

The QoS TCAM region Ingress SUP [ingress-sup] must be set to 768 or higher. Follow these steps:

1. Check the TCAM region by using the **show hardware access-list tcam region** command.
2. If the Ingress SUP region is not set to 768 or higher, then configure the Ingress SUP TCAM region using the command **hardware access-list tcam region ing-sup 768**. Copy the running configuration to the startup configuration (**copy running-config startup-config**) and reload the switch.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Required: feature ptp Example: <pre>switch(config)# feature ptp switch(config)#</pre>	Enables the global PTP feature.
Step 3	Required: ptp profile { default 8275-1 8275-2 } Example: <pre>switch(config)# ptp profile 8275-1 switch(config-ptp-profile)#</pre>	Enables a PTP profile and enters the PTP profile configuration mode. For more information about the commands supported through the profile types in this command, see Note 8275-1 and 8275-2 support the PTP Telecom Profile configuration. For Cisco NX-OS Release 9.3(5), only the Cisco Nexus 93180YC-FX3S switch supports either option in this command.
Step 4	Profile Default: mode { hybrid non-hybrid none } Example: <pre>switch(config)# mode hybrid switch(config-ptp-profile)#</pre>	Configures the PTP operational mode for the switch: hybrid: The SyncE source acts as the PTP source. default: The local/1588 clock acts as the PTP source. Note • This command is automatically configured when the ptp profile command is set. The configuration value cannot be changed. See Step 3, on page 93 for more information.

	Command or Action	Purpose
		The only mode that the 8275-2 profile supports is none.
Step 5	exit Example: <pre>switch(config-ptp-profile)# exit switch(config)#</pre>	Exits the PTP profile configuration mode and returns to the global configuration mode.
Step 6	ptp source ip-address Example: <pre>switch(config)# ptp source 10.10.10.20 switch(config)#</pre>	Configures the source IPv4 address for all the PTP packets in the multicast PTP mode.
Step 7	Profile Default: ptp priority1 value Example: <pre>switch(config)# ptp priority1 128 switch(config)#</pre>	Configures the priority1 value to use when advertising this clock. This value overrides the default criteria (clock quality, clock class, etc.) for best master clock selection. Lower values take precedence. Note This command is automatically configured when the ptp profile 8275-1 or 8275-2 global command is set. The configuration value cannot be changed. See Step 3, on page 93.
Step 8	Profile Default: ptp priority2 value Example: <pre>switch(config)# ptp priority2 128 switch(config)#</pre>	Configures the priority2 value to use when advertising this clock. This value overrides the default criteria (clock quality, clock class, etc.) for best master clock selection. Lower values take precedence. Default: 128 Range: 0 through 255 Note This command is automatically configured when the ptp profile 8275-1 or 8275-2 global command is set. See Step 3, on page 93.
Step 9	ptp pdelay-req-interval value Example: <pre>switch(config)# ptp pdelay-req-interval 0 switch(config)#</pre>	Configures the peer delay request interval. <i>value</i>: Range is from 0 through 5.
Step 10	Profile Default: ptp domain value Example:	Specifies the PTP clock domain value. The allowed domain number range for G.8275.1 profile is between 24 and 43. Default is 24.

	Command or Action	Purpose
	<pre>switch(config)# ptp domain 24 switch(config)#</pre>	The allowed domain number range for G.8275.2 profile is between 44 and 63. Default is 44. Note This command is automatically configured when the ptp profile 8275-1 or 8275-2 global command is set. See Step 3, on page 93.

Configure PTP Telecom Profile 8275-1 on an Interface

This procedure describes the steps that are involved to configure PTP telecom profile 8275-1 for interfaces.



Note Some commands that are described in this procedure are automatically enabled and configured when the **ptp profile 8275-1** global command is set and PTP is enabled on the interface. See [Configure Global PTP Telecom Profile, on page 92](#) for more information.

Before you begin

This procedure along with configuring frequency synchronization on the interface, constitutes the required interface settings for the "hybrid PTP" platform. For more information about the interface frequency synchronization configuration, see [Configuring Frequency Synchronization on an Interface, on page 52](#).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface ethernet <i>slot / port</i> Example: <pre>switch(config)# interface ethernet 1/5 switch(config-if)#</pre>	Specifies the interface on which you are configuring PTP telecom profile parameters and enters the interface configuration mode.
Step 3	[no] ptp Example: <pre>switch(config-if)# ptp switch(config-if)#</pre>	Enables PTP on the interface.

	Command or Action	Purpose
Step 4	Profile Default: ptp transport ethernet Example: <pre>switch(config-if)# ptp transport ethernet switch(config-if)#</pre>	Specifies the transport mechanism that is used to send PTP packets. For ethernet , PTP packets are carried only in Eth frame (Eth/ptp). Note This command is automatically configured when the ptp profile 8275-1 global command is set. For more information about the ptp profile 8275-1 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 5	Profile Default: ptp transmission multicast Example: <pre>switch(config-if)# ptp transmission multicast switch(config-if)#</pre>	Configures the PTP transmission method that is used by the interface. For multicast , PTP uses multicast destination IP address 224.0.1.129 as per IEEE 1588 standards for communication between devices. Note This command is automatically configured when the ptp profile 8275-1 global command is set. For more information about the ptp profile 8275-1 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 6	Profile Default: ptp role dynamic Example: <pre>switch(config-if)# ptp role dynamic switch(config-if)#</pre>	Configures the PTP role of the interface. For dynamic , the best master clock algorithm (BMCA) assigns the role. Note This command is automatically configured when the ptp profile 8275-1 global command is set. For more information about the ptp profile 8275-1 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 7	(Optional) ptp destination-mac non-forwardable rx-no-match accept Example: <pre>switch(config-if)# ptp destination-mac non-forwardable rx-no-match accept switch(config-if)#</pre>	Accepts and responds to nonforwardable destination MAC address packets. These destination MAC addresses are used in the PTP messages that are exchanged between the GM clock, PTP-master clock, and PTP-slave clocks.
Step 8	Profile Default: ptp cost value Example: <pre>switch(config-if)# ptp cost 128 switch(config-if)#</pre>	Configures the value used in the BMCA's selection of the best master clock. If all the parameters mentioned in the standard are the same, then this local priority is used. Note This command is automatically configured when the ptp profile 8275-1 global command is set. For more information about the ptp

	Command or Action	Purpose
		profile 8275-1 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 9	Profile Default: ptp delay-request minimum interval <i>log-seconds</i> Example: <pre>switch(config-if)# ptp delay-request minimum interval -4</pre>	Configures the minimum interval that is allowed between PTP delay messages when the port is in the master state. Note This command is automatically configured when the ptp profile 8275-1 global command is set. For more information about the ptp profile 8275-1 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 10	Profile Default: ptp announce interval <i>log-seconds</i> Example: <pre>switch(config-if)# ptp announce interval -3</pre>	Configures the interval between PTP announce messages on an interface or the number of PTP intervals before a timeout occurs on an interface. Note This command is automatically configured when the ptp profile 8275-1 global command is set. For more information about the ptp profile 8275-1 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 11	Profile Default: ptp sync interval <i>log-seconds</i> Example: <pre>switch(config-if)# ptp sync interval -4</pre>	Configures the interval between PTP synchronization messages on an interface. Note This command is automatically configured when the ptp profile 8275-1 global command is set. For more information about the ptp profile 8275-1 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 12	(Optional) [no] ptp announce timeout count Example: <pre>switch(config-if)# ptp announce timeout 3</pre>	Configures the number of PTP intervals before a timeout occurs on an interface. The range for the PTP announcement interval timeout is from 2 to 4 intervals.
Step 13	(Optional) [no] ptp profile-override Example: <pre>switch(config-if)# ptp profile-override switch(config-if)#</pre>	Disabled by default, when enabled, allows you to change the following commands in this interface configuration: • ptp transport • ptp announce interval • ptp delay-request minimum interval

	Command or Action	Purpose
		• ptp sync interval • ptp cost (8275-1 profile only) Note When enabled, changes to the commands will not be reset to default if the global PTP profile is changed. Removing ptp profile-override resets the PTP configuration on the interface to the default values corresponding to the global profile.

Configure PTP Telecom Profile 8275-2 on an Interface

This procedure describes the steps that are involved to configure PTP telecom profile 8275-2 for interfaces.



Note Some commands that are described in this procedure are automatically enabled and configured when the **ptp profile 8275-2** global command is set and PTP is enabled on the interface. See [Configure Global PTP Telecom Profile, on page 92](#) for more information.

Before you begin

This procedure along with configuring frequency synchronization on the interface, constitutes the required interface settings for the "hybrid PTP" platform. For more information about the interface frequency synchronization configuration, see [Configuring Frequency Synchronization on an Interface, on page 52](#).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface ethernet slot / port Example: <pre>switch(config)# interface ethernet 1/5 switch(config-if)#</pre>	Specifies the interface on which you are configuring PTP telecom profile parameters and enters the interface configuration mode.
Step 3	[no] ptp Example: <pre>switch(config-if)# ptp switch(config-if)#</pre>	Enables PTP on the interface.

	Command or Action	Purpose
Step 4	Profile Default: ptp transport ipv4 Example: <pre>switch(config-if)# ptp transport ipv4 switch(config-if)#</pre>	Specifies the transport mechanism that is used to send PTP packets. Note This command is automatically configured when the ptp profile 8275-2 global command is set. For more information about the ptp profile 8275-2 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 5	Profile Default: ptp transmission unicast Example: <pre>switch(config-if)# ptp transmission unicast switch(config-if)#</pre>	Configures the PTP transmission method that is used by the interface. For unicast , PTP uses the configured master/slave/peer IP. Note This command is automatically configured when the ptp profile 8275-2 global command is set. For more information about the ptp profile 8275-2 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 6	Profile Default: ptp role dynamic Example: <pre>switch(config-if)# ptp role dynamic switch(config-if)#</pre>	Configures the PTP role of the interface. For dynamic , the best master clock algorithm (BMCA) assigns the role. Note This command is automatically configured when the ptp profile 8275-2 global command is set. For more information about the ptp profile 8275-2 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 7	(Optional) ptp destination-mac non-forwardable rx-no-match accept Example: <pre>switch(config-if)# ptp destination-mac non-forwardable rx-no-match accept switch(config-if)#</pre>	Accepts and responds to non-forwardable destination MAC address packets. These destination MAC addresses are used in the PTP messages that are exchanged between the GM clock, PTP-master clock, and PTP-slave clocks.
Step 8	Profile Default: ptp cost value Example: <pre>switch(config-if)# ptp cost 128 switch(config-if)#</pre>	Configures the value used in the BMCA's selection of the best master clock. If all the parameters mentioned in the standard are the same, then this local priority is used. Note This command is automatically configured when the ptp profile 8275-2 global command is set. For more information about the ptp profile 8275-2 command, see Configure Global PTP Telecom Profile, on page 92 .

	Command or Action	Purpose
Step 9	Profile Default: ptp delay-request minimum interval <i>log-seconds</i> Example: <pre>switch(config-if)# ptp delay-request minimum interval -4</pre>	Configures the minimum interval that is allowed between PTP delay messages when the port is in the master state. Note This command is automatically configured when the ptp profile 8275-2 global command is set. For more information about the ptp profile 8275-2 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 10	Profile Default: ptp announce interval <i>log-seconds</i> Example: <pre>switch(config-if)# ptp announce interval -3</pre>	Configures the interval between PTP announce messages on an interface or the number of PTP intervals before a timeout occurs on an interface. Note This command is automatically configured when the ptp profile 8275-2 global command is set. For more information about the ptp profile 8275-2 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 11	Profile Default: ptp sync interval <i>log-seconds</i> Example: <pre>switch(config-if)# ptp sync interval -4</pre>	Configures the interval between PTP synchronization messages on an interface. Note This command is automatically configured when the ptp profile 8275-2 global command is set. For more information about the ptp profile 8275-2 command, see Configure Global PTP Telecom Profile, on page 92 .
Step 12	(Optional) [no] ptp announce timeout count Example: <pre>switch(config-if)# ptp announce timeout 3</pre>	Configures the number of PTP intervals before a timeout occurs on an interface. The range for the PTP announcement interval timeout is from 2 to 4 intervals.
Step 13	(Optional) [no] ptp profile-override Example: <pre>switch(config-if)# ptp profile-override switch(config-if)#</pre>	Disabled by default, when enabled, allows you to change the following commands in this interface configuration: • ptp transport • ptp announce interval • ptp delay-request minimum interval • ptp sync interval • ptp cost (8275-2 profile only)

	Command or Action	Purpose
		Note When enabled, changes to the commands will not be reset to default if the global PTP profile is changed. Removing ptp profile-override resets the PTP configuration on the interface to the default values corresponding to the global profile.

Profile Defaults

The following table lists the ranges and default values for the commands that are automatically configured when the global command **ptp profile** is set. You cannot change the range for the affected global commands beyond those allowed by the configured profile. However, in the interface mode, they can be changed if the **ptp profile-override** command is set.



Note For Cisco NX-OS Release 9.3(5), only the Cisco Nexus 93180YC-FX3S switch supports either option in this command.

Table 6: Range and Default Values

Parameter	Scope or Configuration Mode	Default Profile's Supported Range of Values	Default Profile's Default Value	8275-1 Profile's Supported Range of Values	8275-1 Profile's Default Value	8275-2 Profile's Supported Range of Values	8275-2 Profile's Default Value	With 'ptp profile-override' Configured on an Interface Supported Range of Values (Default is Based on Configured Profile)
mode	global	none	none	hybrid	hybrid	none	none	no change
domain	global	0 to 63	0	24 to 43	24	44 to 63	44	no change
priority1	global	0 to 255	255	128	128	128	128	no change
priority2	global	0 to 255	255	0 to 255	128	0 to 255	128	no change
cost	interface	Not configurable	Not configurable	0 to 255	128	0 to 255	128	0 to 255
transport	interface	ipv4, ipv6	ipv4	ethernet	ethernet	ipv4, ipv6	ipv4	ethernet, ipv4

Parameter	Scope or Configuration Mode	Default Profile's Supported Range of Values	Default Profile's Default Value	8275-1 Profile's Supported Range of Values	8275-1 Profile's Default Value	8275-2 Profile's Supported Range of Values	8275-2 Profile's Default Value	With 'ptp profile-override' Configured on an Interface Supported Range of Values (Default is Based on Configured Profile)
transmission	interface	multicast, unicast	multicast	multicast	multicast	unicast	unicast	no change
role	interface	dynamic, master, slave	dynamic	dynamic	dynamic	dynamic, master, slave	dynamic	no change
announce interval	interface	0 to 4 0 to 4 with aes67 -3 to 1 with smpte-2059-2	1	-3	-3	-3 to 0	-3	-3 to 4 0 to 4 with aes67 -3 to 1 with smpte-2059-2
delay-request minimum interval	interface	-1 to 6 -4 to 5 with aes67 -4 to 5 with smpte-2059-2	0	-4	-4	-4 to 0	-4	-4 to 6 -4 to 5 with aes67 -4 to 5 with smpte-2059-2
sync interval	interface	-3 to 1 -4 to 1 with aes67 -7 to 0 with smpte-2059-2	-2	-4	-4	-4 to 0	-4	-4 to 1 -4 to 1 with aes67 -7 to 0 with smpte-2059-2

Configure PTP Notifications

Before you begin

You can enable, disable, and customize notifications for the following significant PTP events:

- Change in the Grand Master (GM) clock

- Change in the Parent clock
- Change in the PTP state on a port
- High PTP clock corrections

The notifications are generated by the DME infrastructure based on information it receives from PTP.

Procedure

	Command or Action	Purpose
Step 1	[no] ptp notification type gm-change Example: <pre>switch(config)# ptp notification type gm-change switch(config)#</pre>	Configures the system to send a change notification if the PTP grand master clock changes.
Step 2	[no] ptp notification type parent-change Example: <pre>switch(config)# ptp notification type parent-change switch(config)#</pre>	Configures the system to send a change notification if the PTP parent clock changes.
Step 3	[no] ptp notification type port-state-change [category { all master-slave-only }] [interval { immediate seconds [periodic-notification { disable enable }]] Example: <pre>switch(config)# ptp notification type port-state-change category master-slave-only switch(config)#</pre>	Configures the system to send a notification if a port state change event occurs. • category: Specifies which state changes must occur for a notification to be sent. • all: Every port state change is reported. • Note Using the all option results in many notifications. • master-slave-only: Port state changes from and to the master-slave state are only reported. • interval seconds: Port state change notifications are sent at the configured interval: from 1-300 seconds with a granularity of 1 sec. • periodic-notification: Determines if periodic notifications are sent even if a port state change has not occurred during the configured interval. • disable: A port state change notification is reported only if the current state is not the same as the

	Command or Action	Purpose
		previously reported state. Any intermediate state changes during the configured periodic interval are ignored. For example, if a port is a MASTER at time X, and changes to DISABLED and then back to MASTER by the time X+periodic-interval occurs, then no notification is generated for the intervening events. enable: Port state change notifications are sent at the configured interval, irrespective of a change in the port state. • interval immediate: A port State Change Notification is sent when the state changes.
Step 4	[no] ptp notification type high-correction [interval { seconds [periodic-notification { disable enable }] immediate }] Example: <pre>switch(config)# ptp notification type high-correction interval immediate switch(config)#</pre>	Configures the system to send a high-correction notification if a PTP high correction event occurs. A high correction event is when the correction exceeds the value that is configured in the ptp correction-range command (see the following optional step). • interval seconds: High-correction notifications are sent at the configured interval: 1–300 seconds with a granularity of 1 second. • periodic-notification: Determines if periodic notifications are sent even if any high correction has not occurred during the configured interval. • disable: Send a notification only if high correction events occurred during the configured periodic interval. This is the default setting. • enable: Send a notifications irrespective of the number of high correction events during the configured periodic interval. If there are no such events, the payload indicates zero high correction events during the periodic interval. • interval immediate: Send a notification as soon as a high correction event occurs.

	Command or Action	Purpose
		The high correction notification contains the following attributes: • highCorrectionCount • lastHighCorrectionTime • lastHighCorrectionValue
Step 5	(Optional) [no] ptp correction-range { <i>nanoseconds</i> logging } Example: switch(config)# ptp correction-range 200000 switch(config)#	Configures a threshold that, once exceeded, indicates that a PTP high correction has occurred. Range is 10–1000000000. The default is 100000 (100 microseconds).

PTP Mixed Mode

PTP supports Mixed mode for delivering PTP messages, which is detected automatically by Cisco Nexus device, based on the type of **delay_req** message received from connected client and no configuration is required. In this mode when slave sends **delay_req** in unicast message, master also replies with unicast **delay_resp** message.

Configure PTP Interface to Stay in Master State

This procedure describes how to prevent an endpoint from causing a port to transition to a slave state.

Before you begin

- Make sure that you have globally enabled PTP on the switch and configured the source IP address for PTP communication.
- After you globally enable PTP, it is not enabled on all supported interfaces by default. You must enable PTP interfaces individually.

Procedure

	Command or Action	Purpose
Step 1	switch # configure terminal	Enters global configuration mode.
Step 2	switch(config) # interface ethernet slot/port	Specifies the interface on which you are enabling PTP and enters interface configuration mode.
Step 3	switch(config-if) # ptp	Enables or disables PTP on an interface. Note After configuring this command, for Cisco NX-OS Release 9.3(5) and later, skip to step

	Command or Action	Purpose
		5. For Cisco NX-OS Release 9.3(4) and earlier, continue with step 4.
Step 4	switch(config-if) # ptp multicast master-only	Configures the port to maintain the master state. Note This command is supported in Cisco NX-OS Release 9.3(4) and earlier. It is deprecated in Cisco NX-OS Release 9.3(5) and later. For Cisco NX-OS Release 9.3(4) and earlier, this concludes the procedure.
Step 5	ptp role master	Configures the port to maintain the master state. Note This command is supported beginning with Cisco NX-OS Release 9.3(5).

Example

This example shows how to configure PTP on an interface and configure the interface to maintain the Master state:

```
switch(config)# show ptp brief
```

```
PTP port status
```

```
-----
```

```
Port                               State
```

```
-----
```

```
Eth1/1                             Slave
```

```
switch(config)# interface ethernet 1/1
```

```
switch(config-if)# ptp multicast master-only
```

```
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_GM_CHANGE: Grandmaster clock has changed
from 60:73:5c:ff:fe:62:a1:41 to 58:97:bd:ff:fe:0d:54:01 for the PTP protocol
```

```
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_STATE_CHANGE: Interface Eth1/1 change from
PTP_BMC_STATE_SLAVE to PTP_BMC_STATE_PRE_MASTER
```

```
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_TIMESYNC LOST: Lost sync with master clock
```

```
2001 Jan  7 07:50:07 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_STATE_CHANGE: Interface Eth1/1 change from
PTP_BMC_STATE_PRE_MASTER to PTP_BMC_STATE_MASTER
```

Enable PTP Unicast-Negotiation

Enabling PTP unicast transmission is a pre-requisite for using unicast-negotiation.

Beginning with Cisco NX-OS 10.2(1)F release, the following are the newly added CLIs:

Procedure

	Command or Action	Purpose
Step 1	switch (config-ptp-ucast-negotiation)# schema <schema-name>	A default schema will be created when unicast-negotiation is enabled and will be applied to all interfaces that have PTP-unicast enabled and to the master IPs currently configured. Schema name can be any string up to 31 characters.
Step 2	(Optional) switch (config-ptp-ucast-nego-schema)# announce interval <log-seconds>	Configures the interval between PTP announce messages. The range varies from -3 to 0. Default value is -1.
Step 3	(Optional) switch (config-ptp-ucast-nego-schema)# sync interval <log-seconds>	Configures the interval between PTP synchronization messages. The range varies from -4 to 0. Default value is -3.
Step 4	switch (config-ptp-ucast-nego-schema)# delay-response interval <log-seconds>	Configures the interval allowed between PTP delay messages when the port is in master state. The range varies from -4 to 0. Default value is -2.
Step 5	switch (config-ptp-ucast-nego-schema)# announce duration <seconds> [renew-offset <seconds>]	(Optional) Configures duration of announce session. renew-offset<seconds>: This can be used to set how soon the slave sends renewal request for the session. Default value is 10 which means, it will send renewal request 10 seconds before expiry of session (granted duration). The range is 60 to 1000. Default value is 300.
Step 6	switch (config-ptp-ucast-nego-schema)# sync duration <seconds> [renew-offset <seconds>]	(Optional) Configures duration of sync session. renew-offset<seconds>: This can be used to set how soon the slave sends renewal request for the session. Default value is 10 which means, it will send renewal

	Command or Action	Purpose
		request 10 seconds before expiry of session (granted duration). The range is 60 to 1000. Default value is 300.
Step 7	switch (config-ptp-ucast-nego-schema)# delay response duration <seconds> [renew-offset <seconds>]	(Optional) Configures duration of delay-response session. renew-offset <seconds>: This can be used to set how soon the slave sends renewal request for the session. Default value is 10 which means, it will send renewal request 10 seconds before expiry of session (granted duration). The range is 60 to 1000. Default value is 300.
Step 8	switch (config-ptp-ucast-nego-schema)# announce interval range <minimum-log-val> <maximum-log-val>	(Optional) Configures acceptable range of values for announce interval requests from slave. Default for minimum-log-val is -3. Default for maximum-log-val is 0.
Step 9	switch (config-ptp-ucast-nego-schema)# sync interval range <minimum-log-val> <maximum-log-val>	(Optional) Configures acceptable range of values for sync interval requests from slave. Default for minimum-log-val is -4. Default for maximum-log-val is 0.
Step 10	switch (config-ptp-ucast-nego-schema)# delay-response interval range <minimum-log-val> <maximum-log-val>	(Optional) Configures acceptable range of values for delay-response interval requests from slave. Default for minimum-log-val is -4. Default for maximum-log-val is 0.
Step 11	switch (config-ptp-ucast-nego-schema)# announce duration range <minimum-seconds> <maximum-seconds>	(Optional) Configures acceptable range of values for announce session duration requests from slave. Default for minimum-seconds is 60. Default for maximum-seconds is 1000.
Step 12	switch (config-ptp-ucast-nego-schema)# sync duration range <minimum-seconds> <maximum-seconds>	(Optional) Configures acceptable range of values for sync session duration requests from slave. Default for minimum-seconds is 60. Default for maximum-seconds is 1000.

	Command or Action	Purpose
Step 13	switch (config-ptp-ucast-nego-schema)# delay-response duration range <minimum-seconds> <maximum-seconds>	(Optional) Configures acceptable range of values for delay-response session duration requests from slave. Default for minimum-seconds is 60. Default for maximum-seconds is 1000.
Step 14	show ptp unicast-negotiation [interface ethernet slot/port]	Shows the status of unicast-negotiation.

Enhanced Multicast Scale

This feature is to be used only in specific deployment scenarios where higher scaling of PTP multicast secondary devices is required even though the ability to debug is very limited.

This feature has the following limitations:

- The high scale in the number of PTP slaves implies very high PTP control packet rate. As a result, copp rate needs to be increased appropriately. For more information about Configuring Control Plane Policing, refer to the appropriate version of the *Cisco Nexus 9000 Series NX-OS Security Configuration Guide* on cisco.com.
- PTP debugs should be completely disabled using the **no ptp debug all** command, along with various internal logs, for example, PTP internal syslogs. As a result, ability to debug issues will be very limited.
- We recommend that PTP secondary port should not share hardware MAC (port fifo) with any of the scaled PTP primary ports. Additionally, not more than 2 primary ports should be enabled per hardware MAC. The hardware MAC for ports on any given switch can be checked using the following command:
show interface hardware-mappings
- In rare occasions, the corrections can spike to the milliseconds range.

Perform the following command to enable the scaling of PTP multicast secondary devices:

ptp enhanced-client-scale

To view the status of the above command, run the following command:

switch# **show run ptp | grep enhanced**

Timestamp Tagging

The timestamp tagging feature provides precision time information to track in real time when packets arrive at remote devices. Packets are truncated and timestamped using PTP with nanosecond accuracy. Using the TAP aggregation functionality on the switch, along with the Cisco Nexus Data Broker, you can copy the network traffic using SPAN, filter and timestamp the traffic, and send it for recording and analysis.

If you configure **ttag** on an interface, all incoming traffic will be tagged. If you configure **ttag-strip** on an interface all outgoing traffic with ttag will be removed.

Configure Timestamp Tagging



Note Configuring timestamp tagging is not supported on Cisco Nexus 9508 switches with 9636C-R, 9636C-RX, and 9636Q-R line cards.



- Note**
- When you use the ttag feature in a VXLAN EVPN multisite deployment, make sure that the ttag is stripped (**ttag-strip**) on BGW's DCI interfaces that connect to the cloud. To elaborate, if the ttag is attached to non-Nexus 9000 devices that do not support ether-type 0x8905, stripping of ttag is required. If no stripping is done, the non-Nexus devices will drop the packet.
 - BGW back-to-back model of DCI does not require ttag stripping.
 - For Cloudscale platforms, if a packet is received with a TTAG header (ethertype 0x8905), it will be preserved, unless **ttag-strip** is configured on the outgoing interface. After switching or routing, the packet egressing the switch will also have a TTAG header. For this to happen, the switch does not need to have any PTP or TTAG-related configuration. If the next hop would be a non-Cloudscale platform, such as a Catalyst switch - it might not recognize ether-type 0x8905, and could drop the packet.
 - Cisco Nexus 9800 switches do not support routing of ether-type 0x8905 packets.

Before you begin

Make sure that you have globally enabled PTP offloading.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface type slot/port Example: <pre>switch(config)# interface ethernet 2/2 switch(config-if)#</pre>	Enters interface configuration mode for the specified interface.
Step 3	[no] ttag Example: <pre>switch(config-if)# ttag</pre>	Configures timestamp tagging on the Layer 2 or Layer 3 egress interface. This is required on the ingress port for the traffic that needs to be tagged when egressing the switch. This is not required on the egress port.

Configure TTAG Marker Packets and Time Interval

The ttag timestamp field attaches a 48-bit timestamp on the marker packet. This 48-bit timestamp is not a human familiar ASCII based timestamp. To make this 48-bit timestamp human readable, the ttag marker packet can be used to provide additional information to decode the 48-bit timestamp information.

Field	Position (byte:bit)	Length	Definition
Magic		16	By default, this field displays A6A6. This enables to identify ttag-marker packets on the packet stream.
Version		8	Version number. The default version is 1.
Granularity		16	This field represents the granularity of the 48-bit timestamp size. By default, the value is 04, which is 100 picoseconds or 0.1. nanoseconds.
UTc_offset		8	The utc_offset between the ASIC and the UTC clocks. The default value is 0.
Timestamp_hi		32	The high 16-bit of 48- bit ASIC hardware timestamp. Note Add Correction_hi and Correction_lo to Timestamp_hi and Timestamp_lo fields to get the 64-bit ASIC hardware timestamp.
Timestamp_lo		32	The low 32-bit of 48- bit ASIC hardware timestamp. Note Add Correction_hi and Correction_lo to Timestamp_hi and Timestamp_lo fields to get the 64-bit ASIC hardware timestamp.
UTC sec		32	The seconds part of UTC timestamp from the CPU clock of the Cisco Nexus 9000 Series switch.
UTC nsec		32	The nanoseconds part of UTC timestamp from the CPU clock of the Cisco Nexus 9000 Series switch.
Reserved		32	Reserved for future use.

Correction_hi		32	The high 32-bit of cumulative PTP correction on the Cisco Nexus 9000 Series switch. Note Add Correction_hi and Correction_lo to Timestamp_hi and Timestamp_lo fields to get the 64-bit ASIC hardware timestamp.
Correction_lo		32	The low 32-bit of cumulative PTP correction on the Cisco Nexus 9000 Series switch. Note Add Correction_hi and Correction_lo to Timestamp_hi and Timestamp_lo fields to get the 64-bit ASIC hardware timestamp.
Signature		32	The default value is 0xA5A5A5A5. This allows a forward search of marker packet and provide references to the UTC timestamp, so the client software can use that reference UTC to recover the 32-bit hardware timestamp in each packet header.
Pad		8 64	This is align byte to convert the ttag-marker align to 4 byte boundary.

Before you begin

Make sure that you have globally enabled PTP offloading.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	ttag-marker-interval seconds Example: <pre>switch(config-if)# ttag-marker-interval 90</pre>	Configures the seconds that a switch will take to send a ttag-marker packet to the outgoing ports. This is a global setting to the switch. By default, it sends a ttag-marker packet every 60 seconds. The range for seconds is from 1 to 25200.

	Command or Action	Purpose
Step 3	interface <i>type slot/port</i> Example: <pre>switch(config)# interface ethernet 2/2 switch(config-if)#</pre>	Enters interface configuration mode for the specified interface.
Step 4	[no] ttag-marker enable Example: <pre>switch(config-if)# ttag-marker enable</pre>	Sends the ttag-marker packets to the outgoing port.
Step 5	ttag-strip Example: <pre>switch(config-if)# ttag-strip</pre>	Removes TTAG from egress packets on the interface.

Verify PTP Configuration

Use one of the following commands to verify the configuration:

Table 7: PTP Show Commands

Command	Purpose
show ptp brief	Displays the PTP status.
show ptp clock	Displays the properties of the local clock, including clock identity.
show ptp clock foreign-masters-record	Displays the state of foreign masters known to the PTP process. For each foreign master, the output displays the clock identity, basic clock properties, and whether the clock is being used as a grandmaster.
show ptp corrections	Displays the last few PTP corrections.
show ptp counters [all interface ethernet slot/port]	Displays the PTP packet counters for all interfaces or for a specified interface.
show ptp parent	Displays the properties of the PTP parent.
show ptp port interface ethernet slot/port	Displays the status of the PTP port on the switch.
show ptp time-property	Displays the PTP clock properties.
show running-config ptp [all]	Displays the running configuration for PTP.
clear ptp counters [all interface ethernet slot/port]	Clears all PTP messages that are received and transmitted on a specific interface or on all interfaces that has PTP enabled.

Command	Purpose
show ptp detail	Displays the list of all connected peers for each PTP port and indicates whether the role is static or dynamic.

Verify PTP Telecom Profile Configuration

After performing the PTP telecom profile configuration tasks, use this reference to verify the configuration.

show running-config ptp all

The output of this command displays global and interface configurations for PTP telecom profile.

The following is an example of the output of the **show running-config ptp all** command:

```
switch# show running-config ptp all
!Command: show running-config ptp all
!Running configuration last done at: Fri Feb 21 20:09:55 2020
!Time: Fri Feb 21 21:10:19 2020

version 9.3(5) Bios:version 01.00
feature ptp

ptp profile 8275-1
  mode hybrid
ptp source 0.0.0.0
ptp device-type boundary-clock
ptp priority1 128
ptp priority2 10
ptp pdelay-req-interval 0
no ptp notification type parent-change
no ptp notification type gm-change
no ptp notification type high-correction
no ptp notification type port-state-change
ptp correction-range 100000
no ptp correction-range logging
ptp management
ptp mean-path-delay 1000000000
ptp domain 24
ttag-marker-interval 60

interface Ethernet1/1
  ptp
  no ptp profile-override
  ptp destination-mac non-forwardable rx-no-match accept
  ptp transport ethernet
  ptp transmission multicast
  ptp role dynamic
  ptp cost 128
  ptp delay-request minimum interval -4
  ptp announce interval -3
  ptp sync interval -4
  ptp announce timeout 3

interface Ethernet1/6
  ptp
  no ptp profile-override
  ptp destination-mac non-forwardable rx-no-match accept
  ptp transport ethernet
```



```
ptp transmission multicast
ptp role dynamic
ptp cost 128
ptp delay-request minimum interval -4
ptp announce interval -3
ptp sync interval -4
ptp announce timeout 3

interface Ethernet1/7
 ptp
 no ptp profile-override
 ptp destination-mac non-forwardable rx-no-match accept
 ptp transport ethernet
 ptp transmission multicast
 ptp role dynamic
 ptp cost 128
 ptp delay-request minimum interval -4
 ptp announce interval -3
 ptp sync interval -4
 ptp announce timeout 3

interface Ethernet1/8
 ptp
 no ptp profile-override
 ptp destination-mac non-forwardable rx-no-match accept
 ptp transport ethernet
 ptp transmission multicast
 ptp role dynamic
 ptp cost 128
 ptp delay-request minimum interval -4
 ptp announce interval -3
 ptp sync interval -4
 ptp announce timeout 3
```



Note The output of the **show running-config ptp all** command displays a complete list of all the PTP configured interfaces.

show ptp parent

The output of this command displays the properties of a PTP parent.

The following is an example of the output of the **show ptp parent** command:

```
switch# show ptp parent
PTP PARENT PROPERTIES

Parent Clock:
Parent Clock Identity: 10:b3:d6:ff:fe:bf:a8:63
Parent Port Number: 0
Observed Parent Offset (log variance): N/A
Observed Parent Clock Phase Change Rate: N/A

Grandmaster Clock:
Grandmaster Clock Identity: 10:b3:d6:ff:fe:bf:a8:63
Grandmaster Clock Quality:
  Class: 248
  Accuracy: 254
  Offset (log variance): 65535
  Priority1: 128
  Priority2: 10
```

show ptp corrections

The output of this command displays up to the last 2000 correction details for each PTP slave port.

The following is an example of the output of the **show ptp corrections** command:

```
switch# show ptp corrections
PTP past corrections
-----
```

Slave Port	SUP Time	Correction (ns)	MeanPath Delay (ns)
Eth1/3	Thu Feb 20 22:51:02 2020 861523	4	260
Eth1/3	Thu Feb 20 22:51:02 2020 735961	4	260
Eth1/3	Thu Feb 20 22:51:02 2020 610170	4	268
Eth1/3	Thu Feb 20 22:51:02 2020 483106	0	280
Eth1/3	Thu Feb 20 22:51:02 2020 355745	0	280
Eth1/3	Thu Feb 20 22:51:02 2020 229924	-4	268
Eth1/3	Thu Feb 20 22:51:02 2020 104819	-4	268
Eth1/3	Thu Feb 20 22:51:01 2020 979604	8	272

show ptp clock

The output of this command displays the properties of the local clock, including clock identity.

The following is an example of the output of the **show ptp clock** command:

```
switch# show ptp clock
PTP Device Type : boundary-clock
PTP Device Encapsulation : NA
PTP Source IP Address : 0.0.0.0
Clock Identity : 10:b3:d6:ff:fe:bf:a8:63
Clock Domain: 24
Slave Clock Operation : Unknown
Master Clock Operation : Two-step
Slave-Only Clock Mode : Disabled
Number of PTP ports: 35
Priority1 : 128
Priority2 : 10
Clock Quality:
    Class : 248
    Accuracy : 254
    Offset (log variance) : 65535
Offset From Master : 0
Mean Path Delay : 0
Steps removed : 0
Correction range : 100000
MPD range : 1000000000
Local clock time : Wed Feb 26 17:08:34 2020
Hardware frequency correction : NA
PTP Clock state : Free-Run
```

show ptp brief

The output of this command displays the PTP clock state for each configured port.

The following is an example of the output of the **show ptp brief** command:

```
switch# show ptp brief
PTP port status
-----
```

Port	State
Eth1/1	Slave
Eth1/6	Disabled

```

Eth1/7           Disabled
Eth1/8           Disabled
Eth1/10          Master
Eth1/11          Disabled
Eth1/12          Disabled
Eth1/13          Master
Eth1/14          Disabled
Eth1/15          Disabled
Eth1/16          Disabled
Eth1/17          Disabled
Eth1/18          Disabled
Eth1/19          Disabled
Eth1/20          Disabled
Eth1/21          Disabled
Eth1/22          Disabled
Eth1/23          Disabled
Eth1/24          Disabled
Eth1/25          Disabled
Eth1/26          Disabled
Eth1/27          Disabled
Eth1/28          Disabled
Eth1/29          Disabled
Eth1/30          Disabled
Eth1/31          Disabled
Eth1/32          Disabled
Eth1/33          Disabled
Eth1/34          Disabled
Eth1/35          Disabled
Eth1/36          Disabled
Eth1/37          Disabled
Eth1/38          Disabled
Eth1/39          Disabled
Eth1/40          Disabled

```

show ptp clock foreign-masters record

The output of this command displays the state of foreign masters known to the PTP process. For each foreign master, the output displays the clock identity, basic clock properties, and whether the clock is being used as a grandmaster..

The following is an example of the output of the **show ptp clock foreign-master-record** command:

```

switch# show ptp port status
P1=Priority1, P2=Priority2, C=Class, A=Accuracy,
OSLV=Offset-Scaled-Log-Variance, SR=Steps-Removed
GM=Is grandmaster

```

Interface	Clock-ID	P1	P2	C	A	OSLV	SR
Eth1/1	00:00:00:00:00:00:00:01	128	128	6	33	65535	0 GM

show ptp detail

The output of this command displays the list of all connected peers for each PTP port and indicates whether the role is static or dynamic.

```

switch(config-if)# show ptp detail
PTP port details:
  Flags:      (S) Static, (D) Dynamic
  Connected Peer details: IP clock-id

```

Port	State	Transmission	Transport	Flags
====	====	=====	=====	=====
Eth1/27/1	Slave	Unicast	IPV4	(S)
Connected Peers:				
10.10.1.36		00:fd:22:ff:fe:8d:1a:a5-0x1901		
Eth1/27/2	Passive	Unicast	IPV4	(D)
Connected Peers:				
10.10.2.36		00:00:00:00:00:00:00:00-0x0		
Eth1/27/3	Master	Multicast	IPV4	(D)
Connected Peers:				

Configuration Examples

This example shows how to configure PTP globally on the device, specify the source IP address for PTP communications, and configure a preference level for the clock:

```
switch# configure terminal
switch(config)# feature ptp
switch(config)# ptp source 10.10.10.1
switch(config)# ptp priority1 1
switch(config)# ptp priority2 1
switch(config)# show ptp brief
PTP port status
-----
Port State
-----
switch(config)# show ptp clock
PTP Device Type: Boundary clock
Clock Identity : 0:22:55:ff:ff:79:a4:c1
Clock Domain: 0
Number of PTP ports: 0
Priority1 : 1
Priority2 : 1
Clock Quality:
  Class : 248
  Accuracy : 254
  Offset (log variance) : 65535
Offset From Master : 0
Mean Path Delay : 0
Steps removed : 0
Local clock time:Mon Dec 22 14:13:24 2014
```

This example shows how to configure PTP on an interface and configure the intervals for the announce, delay-request, and synchronization messages:

```
switch# configure terminal
switch(config)# interface Ethernet 1/1
switch(config-if)# ptp
switch(config-if)# ptp announce interval 3
switch(config-if)# ptp announce timeout 2
switch(config-if)# ptp delay-request minimum interval smpte-2059-2 -3
switch(config-if)# ptp sync interval smpte-2059-2 -3
switch(config-if)# no shutdown
switch(config-if)# show ptp brief
PTP port status
-----
```

```

Port State
-----
Eth2/1 Master
switch(config-if)# show ptp port interface ethernet 2/1
PTP Port Dataset: Eth2/1
Port identity: clock identity: 0:22:55:ff:ff:79:a4:c1
Port identity: port number: 1028
PTP version: 2
Port state: Master
Delay request interval(log mean): 4
Announce receipt time out: 2
Peer mean path delay: 0
Announce interval(log mean): 3
Sync interval(log mean): 1
Delay Mechanism: End to End
Peer delay request interval(log mean): 0

```

This example shows how to configure master/slave role and assign corresponding peer slave/master IP addresses.

```

For Cisco NX-OS Release 9.3(5) and later:
switch# configure terminal
switch(config)# interface ethernet 1/1
switch(config-if)# ptp
switch(config-if)# ptp transmission unicast
switch(config-if)# ptp role master
switch(config-if)# ptp slave 10.1.1.2
switch(config-if)# ptp ucast-source 11.0.0.1
switch(config-if)# ip address 11.0.0.1/24
switch(config-if)# no shutdown

```

```

switch# configure terminal
switch(config)# interface ethernet 1/1
switch(config-if)# ptp
switch(config-if)# ptp transmission unicast
switch(config-if)# ptp role slave
switch(config-if)# ptp master 10.1.1.2
switch(config-if)# ptp ucast-source 11.0.0.1
switch(config-if)# ip address 11.0.0.1/24
switch(config-if)# no shutdown

```

```

For Cisco NX-OS Release 9.3(4) and earlier:
switch-1(config)# interface ethernet 1/1
switch-1(config-if)# ptp transport ipv4 ucast master
switch-1(config-if-ptp-master)# slave ipv4 1.2.3.1
switch-1(config-if-ptp-master)# slave ipv4 1.2.3.2
switch-1(config-if-ptp-master)# slave ipv4 1.2.3.3
switch-1(config-if-ptp-master)# slave ipv4 1.2.3.4
switch-1(config-if-ptp-master)#

```

```

switch-1(config-if)# ptp transport ipv4 ucast slave
switch-1(config-if-ptp-slave)# master ipv4 4.4.4.1
switch-1(config-if-ptp-slave)# master ipv4 4.4.4.2
switch-1(config-if-ptp-slave)# master ipv4 4.4.4.3

```

```

switch-1(config-if-ptp-slave)# ptp ucast-source 9.9.9.9

```

```

switch-1(config-if)# sh running-config ptp

```

```

!Command: show running-config ptp

```

```

!Time: Tue Feb  7 17:37:09 2017

version 7.0(3)I4(6)
feature ptp

ptp source 1.1.1.1

interface Ethernet1/1
  ptp transport ipv4 ucast master
    slave ipv4 1.2.3.1
    slave ipv4 1.2.3.2
    slave ipv4 1.2.3.3
    slave ipv4 1.2.3.4

interface Ethernet1/2
  ptp transport ipv4 ucast slave
    master ipv4 4.4.4.1
    master ipv4 4.4.4.2
    master ipv4 4.4.4.3
  ptp ucast-source 9.9.9.9

switch-1(config-if)#

```

This example shows how to configure PTP in clock operation mode with master or slave ports.

```

PLTFM-A(config)# show ptp clock
PTP Device Type : boundary-clock
PTP Device Encapsulation : layer-3
PTP Source IP Address : 1.1.1.1
Clock Identity : 74:26:ac:ff:fe:fd:de:ff
Clock Domain: 0
Slave Clock Operation : One-step
Master Clock Operation : One-step
Slave-Only Clock Mode : Disabled
Number of PTP ports: 142
Priority1 : 200
Priority2 : 200
Clock Quality:
  Class : 248
  Accuracy : 254
  Offset (log variance) : 65535
Offset From Master : -32
Mean Path Delay : 105
Steps removed : 1
Correction range : 200
MPD range : 100
Local clock time : Wed Jul  3 18:57:23 2019
Hardware frequency correction : NA

```

Additional References

Related Documents

Related Topic	Document Title
1588 IEEE	1588 IEEE standards



CHAPTER 7

Configuring GPS

This chapter describes the GPS input and how to configure it.

This chapter includes the following sections:

- [About GPS, on page 121](#)
- [Guidelines and Limitations for GPS, on page 121](#)
- [Configuring GPS Settings for the Grandmaster Clock, on page 122](#)
- [Verifying the GPS Configuration, on page 123](#)

About GPS

The router can receive 1PPS, 10 MHz, and ToD signals from an external clocking and timing source. The three inputs are combined as a Sync-2 interface to form the external timing source or the GPS input.

The GPS front panel connector details are:

- ToD—RS422 format as input
- 1PPS—RS422 or DIN connector as input
- 10MHz—DIN connector as input

GPS input starts only when all the three signals – 1PPS, 10MHz, and ToD are UP.



Note Unlike the Ethernet interface, the Sync-2 interface cannot receive or transmit QL. Ensure that you assign a QL value to the Sync-2 interface.

By default, 1PPS and 10MHz are in output mode. ToD output mode is not configurable.

Guidelines and Limitations for GPS

GPS has the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 10.3(2)F, the GPS input is supported only on the Cisco Nexus 93180YC-FX3 switch with following limitations:

- The ToD output is not supported.
- RS422 ToD input is not supported.
 - GPS without RS422 TOD input can be used only for frequency and phase synchronization.

Configuring GPS Settings for the Grandmaster Clock

Use this procedure to enable the GPS as input.

Procedure

	Command or Action	Purpose												
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.												
Step 2	clock-interface sync 1/1 Example: <pre>switch(config)# clock-interface sync 1/1 switch(config-clk-if)# port-parameters switch(config-clk-params)#</pre>	Enter the GPS clock configuration mode.												
Step 3	gps-input tod-format value pps-input value Example: <pre>switch(config-clk-params)# gps-input tod-format ntp4 pps-input ttl switch(config-clk-params)# exit switch(config-clk-if)#</pre>	Enables the GPS as input and configures the source for PPS input and format for the ToD input. Supported ToD formats are: <table><tr><th>Incoming ToD Input</th><th>Format</th></tr><tr><td>cisco</td><td>ASCII format</td></tr><tr><td>ntp4</td><td>NTP Type4 format</td></tr></table> Supported PPS input formats are: <table><tr><th>Incoming PPS Input</th><th>Format</th></tr><tr><td>RS422</td><td>RS422 format</td></tr><tr><td>TTL</td><td>TTL format</td></tr></table>	Incoming ToD Input	Format	cisco	ASCII format	ntp4	NTP Type4 format	Incoming PPS Input	Format	RS422	RS422 format	TTL	TTL format
Incoming ToD Input	Format													
cisco	ASCII format													
ntp4	NTP Type4 format													
Incoming PPS Input	Format													
RS422	RS422 format													
TTL	TTL format													

	Command or Action	Purpose
Step 4	frequency synchronization Example: <pre>switch(config-clk-if)# frequency synchronization switch(config-clk-freqsync)# selection input switch(config-clk-freqsync)# wait-to-restore 0 switch(config-clk-freqsync)# exit switch(config-clock-if)#</pre>	Configures the frequency synchronization on GPS.

Verifying the GPS Configuration

After performing the GPS configuration tasks, use this reference to check for configuration errors and to verify the configuration.

show clock-interface

The output of this command displays details of the GPS clock configuration.

The following is an example of the output of the **show frequency synchronization** selection command:

```
switch(config-clk-if)# sh frequency synchronization selection
=====
Selection point: System Clock (T0) Selector (2 inputs, 1 selected)
Last programmed 00:53:56 ago, and selection made 00:53:35 ago
Next selection points
Node scoped :
Uses frequency selection
Used for local line interface output
S Input Last Selection Point QL Pri Status
== =====
Internal0[1] n/a SEC 255 Available
11 Sync1[1] n/a PRC 100 Locked
=====
Selection point: IEEE 1588 Clock Selector (2 inputs, 1 selected)
Last programmed 00:53:56 ago, and selection made 00:53:55 ago
Next selection points
Node scoped :
Uses frequency selection
S Input Last Selection Point QL Pri Status
== =====
21 Internal0[1] n/a SEC 255 Holdover
Sync1[1] n/a PRC 100 Unmonitored
=====
switch(config-clk-if)#
```

When the front panel timing LED is Green, it indicates that the GPS is configured and 1PPS, ToD, and 10M inputs are valid.

Following is the default LED status for GPS Timing based on the input to the device:

TIMING	Off	GPS config and GPS port is down. Time-of-day (ToD), 1PPS, and 10-MHz ports are not provisioned or disabled.
	Amber	ToD, 1PPS, and 10-MHz signals are not valid.
	Green	GPS port is up. ToD, 1PPS, and 10-MHz signals are valid.



CHAPTER 8

Configuring GNSS

This chapter describes how to configure the Global Navigation Satellite System (GNSS) on Cisco NX-OS devices.

This chapter includes the following sections:

- [About GNSS, on page 125](#)
- [Guidelines and Limitations for GNSS, on page 126](#)
- [Enabling the GNSS Receiver, on page 126](#)
- [Verifying the GNSS Configuration, on page 128](#)

About GNSS

The GNSS receiver is designed to operate on the GPS, Galileo, GLONASS, BeiDou and QZSS L1 frequencies 1551MHz to 1614MHz, standard position service, and Coarse Acquisition code. When connected to an external GNSS antenna, the receiver contains all the circuitry necessary to automatically acquire GNSS satellite signals, track up to 32 GNSS satellites, and compute location, speed, heading, and time. It provides an accurate one pulse-per-second (PPS) and stable 10-MHz frequency output for internal system use.

It automatically initiates a self-survey upon acquisition of GNSS satellites. When the survey is completed, the receiver switches into the “Over-Determined” timing mode. In this mode, the reference position from the self-survey is maintained in memory and the receiver solves only for clock error and clock bias. The receiver provides for both Position and Time Receiver Autonomous Integrity Monitoring (T-RAIM) which allows the receiver to self-determine position change or to remove a satellite providing incorrect information to the timing solution.



Note No signals are transmitted out of GNSS ports (Antenna port).

The Global Navigation Satellite System (GNSS) supports various satellite constellations, each with specific Pseudo-Random Noise (PRN) numbers. The supported PRN ranges for different GNSS systems are:

- GPS – Supports PRN 1 through 32 only
- GLONASS – Supports PRN 65 through 96 only
- BeiDou – Supports PRN 201 through 237 only
- Galileo – Supports PRN 97 through 133 only

- QZSS – Supports PRN 193 through 195 only
- SBAS – Supports PRN 33 through 54 only

Guidelines and Limitations for GNSS

GNSS has the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 10.3(2)F, the GNSS is supported on the Cisco Nexus 93180YC-FX3 switch.

Enabling the GNSS Receiver

Use this procedure to enable the GNSS Receiver as input.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	feature frequency-synchronization Example: switch# feature frequency-synchronization switch(config)#	Enable the feature frequency-synchronization.
Step 3	gnss-receiver sync 1/2 Example: switch(config)# gnss-receiver sync 1/2 switch(config-gnss-if)# no shutdown switch(config-gnss-if)#	Enable the GNSS Receiver as input.
Step 4	frequency synchronization Example: switch(config-gnss-if)# frequency synchronization switch(config-gnss-freqsync)# selection input switch(config-gnss-freqsync)# wait-to-restore 0 switch(config-gnss-freqsync)# exit	Configures the frequency synchronization on GNSS Receiver.
Step 5	(Optional) constellation <type> Example:	You can configure the GNSS module to automatically track any satellite or configure it to explicitly use a specific constellation as

	Command or Action	Purpose
	<pre>switch(config-gnss)# constellation gps switch(onfig-gnss)#</pre>	mentioned in the below table. The default constellation configuration is AUTO: • Auto • GPS • GALILEO • BEIDOU • QZSS • GLONASS
Step 6	(Optional) cable-delay compensation Example: <pre>switch(config-gnss)# cable-delay compensation 0 switch(onfig-gnss)#</pre>	Configures the delay that is introduced by the antenna cable. For long cable runs, this delay can be significant. The range is -1000000, +1000000 nanoseconds. Note The cable delay is about 5.9 nanoseconds per meter of cable.
Step 7	(Optional) elevation threshold Example: <pre>switch(config-gnss)# elevation threshold 10 switch(onfig-gnss)#</pre>	Configures the lowest satellite elevation for fixes. The range is from 0 to 90°. The default value is 10°.
Step 8	(Optional) snr threshold Example: <pre>switch(config-gnss)# snr threshold 10 switch(onfig-gnss)#</pre>	Configures the Carrier to Noise ratio (C/No). The range is 0 to 15db-Hz. The default value is 0.0 C/No. Note Applicable only in Over-Determined Clock. Not applicable during selfsurvey.
Step 9	(Optional) pdop threshold Example: <pre>switch(config-gnss)# pdop threshold 6 switch(onfig-gnss)#</pre>	Position Dilution of Precision (PDOP) indicates the confidence level of a position fix. Low DOP values indicate a high confidence level, while high DOP values indicate a low confidence level. The range is from 0 to 10. The default value is 6.
Step 10	(Optional) 1pps polarity Example: <pre>switch(config-gnss)# 1pps polarity positive switch(onfig-gnss)#</pre>	Configures the polarity of the GNSS receiver. It can be set to either positive or negative. The default option is positive.

	Command or Action	Purpose
Step 11	(Optional) anti-jam disable Example: <pre>switch(config-gnss)# anti-jam disable switch(onfig-gnss)#</pre>	Enables or disables the jamming occurred when the receiver function is disturbed by external RF sources that interfere with GNSS signals or saturate the antenna LNA or receiver front end. Values are Enable or Disable. Note If enabled, minimum of two satellites required for fix in Over-Determined Clock mode.

Verifying the GNSS Configuration

After performing the GNSS configuration tasks, use this reference to check for configuration errors and verify the configuration.

show gnss-receiver

The output of this command displays details of GNSS configuration.

The following is an example of the output of the **show gnss-receiver** command:

```
switch(config-gnss-if)# sh gnss-receiver
GNSS-receiver SYNC 01/02
Status: Available, Up
Position: 41:11:660 N 74:0:0 W -12.805 m
Time: 1648046422 (UTC offset: 18s)
Firmware version: 1.7
Lock Status: Phase Locked, Receiver Mode: 3D-fix
Survey Progress: 100, Holdover Duration: 0
Major Alarm: Not used
Minor Alarm: Antenna shorted
Anti-jam: Disabled, Cable-delay compensation: 1000
1PPS polarity: Negative
PDOP: 0.000, HDOP: 0.000, VDOP: 0.000, TDOP: 0.001
Constellation: Auto, Satellite Count: 7
Satellite Thresholds:
SNR - 0 dB-Hz, Elevation - 0 degrees, PDOP - 5, TRAIM - 1 us
Satellite Info:
PRN    Channel Acquisition Ephemeris  SV      Signal
No.    No.      Flag      Flag      Type    Strength  Elevat'n  Azimuth
-----
11      0        On        On        GPS     0.036    0.076    0.024
28      1        On        On        GPS     0.036    0.025    0.272
1       2        On        On        GPS     0.037    0.089    0.002
19      3        On        On        GPS     0.037    0.036    0.151
14      5        On        On        GPS     0.036    0.019    0.045
17      6        On        On        GPS     0.037    0.025    0.314
23      7        On        On        GPS     0.037    0.014    0.178
switch(config-gnss-if)#
```

show frequency synchronization selection

The following is an example of the output of the **show frequency synchronization selection** command:

```
switch(config-gnss-if)# sh frequency synchronization selection
=====
```

```
Selection point: System Clock (T0) Selector (2 inputs, 1 selected)
Last programmed 00:53:56 ago, and selection made 00:53:35 ago
Next selection points
Node scoped :
Uses frequency selection
Used for local line interface output
S Input Last Selection Point QL Pri Status
== =====
Internal0[1] n/a SEC 255 Available
11 GNSS2[1] n/a PRC 100 Locked
=====
Selection point: IEEE 1588 Clock Selector (2 inputs, 1 selected)
Last programmed 00:53:56 ago, and selection made 00:53:55 ago
Next selection points
Node scoped :
Uses frequency selection
S Input Last Selection Point QL Pri Status
== =====
21 Internal0[1] n/a SEC 255 Holdover
GNSS2[1] n/a PRC 100 Unmonitored
=====
switch(config-gnss-if)#
```




CHAPTER 9

Configuring NTP

This chapter describes how to configure the Network Time Protocol (NTP) on Cisco NX-OS devices.

This chapter includes the following sections:

- [About NTP, on page 131](#)
- [Prerequisites for NTP, on page 133](#)
- [Guidelines and Limitations for NTP, on page 133](#)
- [Default Settings for NTP, on page 135](#)
- [Configuring NTP, on page 135](#)
- [Verifying the NTP Configuration, on page 144](#)
- [Configuration Examples for NTP, on page 144](#)
- [Additional References, on page 146](#)

About NTP

The Network Time Protocol (NTP) synchronizes the time of day among a set of distributed time servers and clients so that you can correlate events when you receive system logs and other time-specific events from multiple network devices. NTP uses the User Datagram Protocol (UDP) as its transport protocol. All NTP communications use Coordinated Universal Time (UTC).

An NTP server usually receives its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server, and then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of each other.

NTP uses a stratum to describe the distance between a network device and an authoritative time source:

- A stratum 1 time server is directly attached to an authoritative time source (such as a radio or atomic clock or a GPS time source).
- A stratum 2 NTP server receives its time through NTP from a stratum 1 time server.

Before synchronizing, NTP compares the time reported by several network devices and does not synchronize with one that is significantly different, even if it is a stratum 1. Because Cisco NX-OS cannot connect to a radio or atomic clock and act as a stratum 1 server, we recommend that you use the public NTP servers available on the Internet. If the network is isolated from the Internet, Cisco NX-OS allows you to configure the time as though it were synchronized through NTP, even though it was not.



Note You can create NTP peer relationships to designate the time-serving hosts that you want your network device to consider synchronizing with and to keep accurate time if a server failure occurs.

The time kept on a device is a critical resource, so we strongly recommend that you use the security features of NTP to avoid the accidental or malicious setting of incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

NTP Associations

An NTP association can be one of the following:

- A peer association—The device can either synchronize to another device or allow another device to synchronize to it.
- A server association—The device synchronizes to a server.

You need to configure only one end of an association. The other device can automatically establish the association.

NTP as a Time Server

The Cisco NX-OS device can use NTP to distribute time. Other devices can configure it as a time server. You can also configure the device to act as an authoritative NTP server, enabling it to distribute time even when it is not synchronized to an outside time source.

Clock Manager

Clocks are resources that need to be shared across different processes. Multiple time synchronization protocols, such as NTP, might be running in the system.

The clock manager allows you to specify the protocol to control the various clocks in the system. Once you specify the protocol, the system clock starts updating. For information on configuring the clock manager, see the [Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide](#).

High Availability

Stateless restarts are supported for NTP. After a reboot or a supervisor switchover, the running configuration is applied. For more information on high availability, see the [Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide](#).

You can configure NTP peers to provide redundancy in case an NTP server fails.

Virtualization Support

NTP recognizes virtual routing and forwarding (VRF) instances. NTP uses the default VRF if you do not configure a specific VRF for the NTP server and NTP peer. See the [Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide](#) for more information about VRFs.

Prerequisites for NTP

NTP has the following prerequisites:

- To configure NTP, you must have connectivity to at least one server that is running NTP.

Guidelines and Limitations for NTP

NTP has the following configuration guidelines and limitations:

- NTP server functionality is supported.
- Before configuring a name based NTP server (FQDN) in a non-default VRF, you must configure a DNS server under that specific VRF. If you configure the DNS server from the global configuration mode using **use-vrf** option, then that name based NTP server configuration will not be added to the running configuration. If you attempted to configure NTP server using this method, you must remove the NTP configuration using the **no** version of the command, add the DNS server under that VRF, and then add name based NTP server to the VRF. The configured DNS server must be reachable and must return the correct IP for the FQDN of the NTP server when queried.
- We recommend that you configure a peer association with another device only when you are sure that your clock is reliable (which means that you are a client of a reliable NTP server).
- A peer that is configured alone takes on the role of a server and should be used as a backup. If you have two servers, you can configure several devices to point to one server and the remaining devices to point to the other server. You can then configure a peer association between these two servers to create a more reliable NTP configuration.
- If you have only one server, we recommend that you configure all the devices as clients to that server.
- You can configure up to 64 NTP entities (servers and peers).
- If you configure NTP in a VRF, ensure that the NTP server and peers can reach each other through the configured VRFs.
- Manually distribute NTP authentication keys on the NTP server and Cisco NX-OS devices across the network.
- If you are using the switch as an edge device and want to use NTP, we recommend using the **ntp access-group** command and filtering NTP only to the required edge devices.
- If the system has been configured with the **ntp passive**, **ntp broadcast client**, or **ntp multicast client** commands, when NTP receives an incoming symmetric active, broadcast, or multicast packet, it can set up an ephemeral peer association in order to synchronize with the sender.



Note

Make sure that you specify **ntp authenticate** before enabling any of the preceding commands. Failure to do so will allow your device to synchronize with any device that sends one of the preceding packet types, including malicious attacker-controlled devices.

- If you specify the **ntp authenticate** command, when a symmetric active, broadcast, or multicast packet is received, the system does not synchronize to the peer unless the packet carries one of the authentication keys that are specified in the **ntp trusted-key** global configuration command.
- To prevent synchronization with unauthorized network hosts, the **ntp authenticate** command should be specified any time the **ntp passive**, **ntp broadcast client**, or **ntp multicast client** command has been specified unless other measures, such as the **ntp access-group** command, have been taken to prevent unauthorized hosts from communicating with the NTP service on the device.
- The **ntp authenticate** command does not authenticate peer associations that are configured via the **ntp server** and **ntp peer** configuration commands. To authenticate the **ntp server** and **ntp peer** associations, specify the **key** keyword.
- A maximum of four IP ACLs can be configured for a single NTP access group. IPv4 and IPv6 ACLs are supported.
- If packet flooding occurs on the inband ports, it can increase the CPU usage by NTPD to more than 90%. To overcome this high CPU usage by NTPD, use the custom CoPP policy to rate limit the incoming traffic to NTP. For more information about creating a custom CoPP policy, refer to the Configuring Control Plane Policing chapter in the relevant version of the Cisco Nexus 9000 Series NX-OS Security Configuration Guide on cisco.com.



Note The recommended rate limit is 1000 kbps for the policy **CIR** field and 64,000 bytes for the **BC** field.

- Beginning with Cisco NX-OS Release 10.1(1), Cisco Nexus 9000 switches do not sync with stratum 14 and 15.
- Beginning with Cisco NX-OS Release 10.1(1), NTP version 4 (NTPv4) is supported on Nexus standalone switches.
- Beginning with Cisco NX-OS Release 10.3(3)F, in line with the RFC 8573 standards, NTP security is enhanced with the AES128CMAC authentication mechanism along with Type-6 encryption support for authentication keys. The following guidelines and limitations are applicable:
 - This feature provides an option to set the password as Type-0, Type-7, or Type-6.
 - The maximum number of unique keys which can be configured is 1024, and the range is 1 to 65535.
 - For Type-6 authentication to work, configure the same primary (master) key on the device, which was used to generate the new Type-6 key being configured, along with **feature password encryption aes**.
 - If you enforce re-encryption using the **encryption re-encrypt obfuscated** command, all the NTP non-Type-6 passwords will be re-encrypted to Type-6.
 - The **encryption delete type6** command deletes all the Type-6 passwords configured on NTP.
 - The **encryption decrypt type6** command decrypts the existing configured Type-6 passwords.
 - To perform ISSD from AES128CMAC/Type-6 supported version to a non-AES128CMAC/Type-6 supported version, unconfigure the Type-6 keys and then perform the ISSD.

- It is recommended to specify the `encryptType` and `keyString` when programmatically (restconf/Netconf and so on) configuring a key chain. If not specified, then the key chain infra uses the already available (or default) value of the missing property to configure the `keyString`.
- If you need to configure with a property missing, then you need to follow the same sequence of steps in both the peer routers.

Default Settings for NTP

The following table lists the default settings for NTP parameters.

Parameters	Default
NTP	Enabled
NTP authentication	Disabled
NTP access	Enabled
NTP logging	Disabled

Configuring NTP



Note Be aware that the Cisco NX-OS commands for this feature may differ from those commands used in Cisco IOS.

Enabling or Disabling NTP

You can enable or disable NTP. NTP is enabled by default.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	[no] feature ntp Example: <code>switch(config)# feature ntp</code>	Enables or disables NTP.
Step 3	(Optional) copy running-config startup-config Example:	Copies the running configuration to the startup configuration.

	Command or Action	Purpose
	<code>switch(config)# copy running-config startup-config</code>	

Configuring the Device as an Authoritative NTP Server

You can configure the device to act as an authoritative NTP server, enabling it to distribute time even when it is not synchronized to an existing time server.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	[no] ntp master [stratum] Example: <code>switch(config)# ntp master</code>	Configures the device as an authoritative NTP server. You can specify a different stratum level from which NTP clients get their time synchronized. The range is from 1 to 15.
Step 3	(Optional) show running-config ntp Example: <code>switch(config)# show running-config ntp</code>	Displays the NTP configuration.
Step 4	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Configuring an NTP Server and Peer

You can configure an NTP server and peer.

Before you begin

Make sure you know the IP address or Domain Name System (DNS) names of your NTP server and its peers.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	[no] ntp server {<i>ip-address</i> <i>ipv6-address</i> <i>dns-name</i>} [key <i>key-id</i>] [maxpoll <i>max-poll</i>] [minpoll <i>min-poll</i>] [prefer] [use-vrf <i>vrf-name</i>] Example: <pre>switch(config)# ntp server 192.0.2.10</pre>	Forms an association with a server. Use the key keyword to configure a key to be used while communicating with the NTP server. The range for the <i>key-id</i> argument is from 1 to 65535. Use the maxpoll and minpoll keywords to configure the maximum and minimum intervals in which to poll a server. The range for the <i>max-poll</i> and <i>min-poll</i> arguments is from 4 to 16 (configured as powers of 2, so effectively 16 to 65536 seconds), and the default values are 6 and 4, respectively (<i>maxpoll</i> default = 64 seconds, <i>minpoll</i> default = 16 seconds). Use the prefer keyword to make this server the preferred NTP server for the device. Use the use-vrf keyword to configure the NTP server to communicate over the specified VRF. The <i>vrf-name</i> argument can be default, management, or any case-sensitive, alphanumeric string up to 32 characters. Note If you configure a key to be used while communicating with the NTP server, make sure that the key exists as a trusted key on the device.
Step 3	[no] ntp peer {<i>ip-address</i> <i>ipv6-address</i> <i>dns-name</i>} [key <i>key-id</i>] [maxpoll <i>max-poll</i>] [minpoll <i>min-poll</i>] [prefer] [use-vrf <i>vrf-name</i>] Example: <pre>switch(config)# ntp peer 2001:0db8::4101</pre>	Forms an association with a peer. You can specify multiple peer associations. Use the key keyword to configure a key to be used while communicating with the NTP peer. The range for the <i>key-id</i> argument is from 1 to 65535. Use the maxpoll and minpoll keywords to configure the maximum and minimum intervals in which to poll a server. The range for the <i>max-poll</i> and <i>min-poll</i> arguments is from 4 to 17 (configured as powers of 2, so effectively 16 to 131072 seconds), and the default values are 6 and 4, respectively (<i>maxpoll</i> default = 64 seconds, <i>minpoll</i> default = 16 seconds). Use the prefer keyword to make this peer the preferred NTP peer for the device.

	Command or Action	Purpose
		Use the use-vrf keyword to configure the NTP peer to communicate over the specified VRF. The <i>vrf-name</i> argument can be default , management , or any case-sensitive, alphanumeric string up to 32 characters.
Step 4	(Optional) show ntp peers Example: <pre>switch(config)# show ntp peers</pre>	Displays the configured server and peers. Note A domain name is resolved only when you have a DNS server configured. When DNS/Name Server resolves both IPv4 and IPv6, IPv6 Address is preferred by NX-OS.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring NTP Authentication

You can configure the device to authenticate the time sources to which the local clock is synchronized. When you enable NTP authentication, the device synchronizes to a time source only if the source carries one of the authentication keys specified by the **ntp trusted-key** command. The device drops any packets that fail the authentication check and prevents them from updating the local clock. NTP authentication is disabled by default.

Before you begin

Make sure that you configured the NTP server with the authentication keys that you plan to specify in this procedure.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] ntp authentication-key number {md5 aes128cmac} password string encryption-type Example: <pre>switch(config)# ntp authentication-key 42 md5 aNiceKey</pre>	Defines the authentication keys. The range for authentication keys is from 1 to 65535. The device does not synchronize to a time source unless the source has one of these authentication keys and the key number is

	Command or Action	Purpose
	<pre>switch(config)# ntp authentication-key 21 md5 JDYk3pp/Fuv0zWyVSRhS6EDERSSpluA7s57dvdsx g74ndf021EI9dF6WX6Z78/5R8qPmSRrRDUDtCcU1Z XDUrF0ErodS3ikPQA= 6 switch(config)# ntp authentication-key 12 aes128cmac JDYkzj4NojJdSkQPvBhFvAO9xCSVwj2iRGvShNSg ER4JwMBMtUEibfqkscgZ4+/iTdDmeCRW9SGWLxKb 3Xk5g8pz4bR7Iula7QA= 6</pre>	specified by the ntp trusted-key number command. You can choose either md5 or aes128cmac authentication scheme. If the user is using the Type-6 key generated from the same primary (master) key, then the device does not synchronize to a time source until the user enables feature password encryption aes. For Type-0 and Type-7 encryption types, the maximum length is 32 characters. Until Release 10.3(3)F, it was 15 (alphanumeric) characters. For the Type-6 encryption type, the maximum limit is 128 characters.
Step 3	ntp server ip-address key key-id Example: <pre>switch(config)# ntp server 192.0.2.1 key 1001</pre>	Forms an association with a server. Use the key keyword to configure a key to be used while communicating with the NTP server. The range for the <i>key-id</i> argument is from 1 to 65535. To require authentication, the key keyword must be used. Any ntp server or ntp peer commands that do not specify the key keyword will continue to operate without authentication.
Step 4	(Optional) show ntp authentication-keys Example: <pre>switch(config)# show ntp authentication-keys</pre>	Displays the configured NTP authentication keys.
Step 5	[no] ntp trusted-key number Example: <pre>switch(config)# ntp trusted-key 42</pre>	Specifies one or more keys (defined in Step 2) that an unconfigured remote symmetric, broadcast, and multicast time source must provide in its NTP packets in order for the device to synchronize to it. The range for trusted keys is from 1 to 65535. This command provides protection against accidentally synchronizing the device to a time source that is not trusted.
Step 6	(Optional) show ntp trusted-keys Example: <pre>switch(config)# show ntp trusted-keys</pre>	Displays the configured NTP trusted keys.

	Command or Action	Purpose
Step 7	[no] ntp authenticate Example: <code>switch(config)# ntp authenticate</code>	Enables or disables authentication for ntp passive, ntp broadcast client, and ntp multicast. NTP authentication is disabled by default.
Step 8	(Optional) show ntp authentication-status Example: <code>switch(config)# show ntp authentication-status</code>	Displays the status of NTP authentication.
Step 9	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Configuring NTP Access Restrictions

You can control access to NTP services by using access groups. Specifically, you can specify the types of requests that the device allows and the servers from which it accepts responses.

If you do not configure any access groups, NTP access is granted to all devices. If you configure any access groups, NTP access is granted only to the remote device whose source IP address passes the access list criteria.

- Without the **match-all** keyword, the packet gets evaluated against the access groups (in the order mentioned below) until it finds a permit. If a permit is not found, the packet is dropped.
- With **match-all** keyword, the packet gets evaluated against all the access groups (in the order mentioned below) and the action is taken based on the last successful evaluation (the last access group where an ACL is configured).
- **peer**—process client, symmetric active, symmetric passive, serve, control, and private packets(all types)
- **serve**—process client, control, and private packets
- **serve-only**—process client packets only
- **query-only**—process control and private packets only

The access groups are evaluated in the following order:

1. **peer** (all packet types)
2. **serve** (client, control, and private packets)
3. **serve-only** (client packets) or **query-only** (control and private packets)

ACL processing of serve-only or query-only depends on the NTP packet type.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] ntp access-group match-all {{peer serve serve-only query-only} access-list-name} Example: <pre>switch(config)# ntp access-group match-all switch(config)# ntp access-group peer peer-acl switch(config)# ntp access-group serve serve-acl</pre>	Creates or removes an access group to control NTP access and applies a basic IP access list. ACL processing stops and does not continue to the next access group option if NTP matches a deny ACL rule in a configured peer. • The peer keyword enables the device to receive time requests and NTP control queries and to synchronize itself to the servers specified in the access list. • The serve keyword enables the device to receive time requests and NTP control queries from the servers specified in the access list but not to synchronize itself to the specified servers. • The serve-only keyword enables the device to receive only time requests from servers specified in the access list. • The query-only keyword enables the device to receive only NTP control queries from the servers specified in the access list. • The match-all keyword enables the access group options to be scanned in the following order, from least restrictive to most restrictive: peer, serve, serve-only, query-only. If the incoming packet does not match the ACL in the peer access group, it goes to the serve access group to be processed. If the packet does not match the ACL in the serve access group, it goes to the serve-only access group, and so on. Note The match-all keyword is available beginning with Cisco NX-OS Release 7.0(3)I6(1) and is supported on Cisco Nexus 9000 Series switches and the Cisco

	Command or Action	Purpose
		Nexus 3164Q, 31128PQ, 3232C, and 3264Q switches. • The <i>access-list-name</i> variable is the name of the NTP access group. The name can be an alphanumeric string up to 64 characters, including special characters.
Step 3	(Optional) show ntp access-groups Example: switch(config)# show ntp access-groups	Displays the NTP access group configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the NTP Source IP Address

NTP sets the source IP address for all NTP packets based on the address of the interface through which the NTP packets are sent. You can configure NTP to use a specific source IP address.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ntp source ip-address Example: switch(config)# ntp source 192.0.2.1	Configures the source IP address for all NTP packets. The <i>ip-address</i> can be in IPv4 or IPv6 format.
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the NTP Source Interface

You can configure NTP to use a specific interface.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ntp source-interface <i>interface</i> Example: switch(config)# ntp source-interface ethernet 2/1	Configures the source interface for all NTP packets. Use the ? keyword to display a list of supported interfaces.
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring NTP Logging

You can configure NTP logging in order to generate system logs with significant NTP events. NTP logging is disabled by default.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] ntp logging Example: switch(config)# ntp logging	Enables or disables system logs to be generated with significant NTP events. NTP logging is disabled by default.
Step 3	(Optional) show ntp logging-status Example: switch(config)# show ntp logging-status	Displays the NTP logging configuration status.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verifying the NTP Configuration

To display the NTP configuration, perform one of the following tasks:

Command	Purpose
show ntp access-groups	Displays the NTP access group configuration.
show ntp authentication-keys	Displays the configured NTP authentication keys.
show ntp authentication-status	Displays the status of NTP authentication.
show ntp logging-status	Displays the NTP logging status.
show ntp peer-status	Displays the status for all NTP servers and peers.
show ntp peers	Displays all the NTP peers.
show ntp rts-update	Displays the RTS update status.
show ntp source	Displays the configured NTP source IP address.
show ntp source-interface	Displays the configured NTP source interface.
show ntp statistics {io local memory peer {ipaddr {ipv4-addr ipv6-addr} name peer-name}}	Displays the NTP statistics.
show ntp trusted-keys	Displays the configured NTP trusted keys.
show running-config ntp	Displays NTP information.

Use the **clear ntp session** command to clear the NTP sessions.

Use the **clear ntp statistics** command to clear the NTP statistics.

Configuration Examples for NTP

This example for md5 shows how to configure the device to synchronize only to time sources that provide authentication key 42 in their NTP packets:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp authentication-key 42 md5 aNiceKey
switch(config)# ntp server 192.0.2.105 key 42
switch(config)# ntp trusted-key 42
switch(config)# ntp authenticate
switch(config)# copy running-config startup-config
[#####] 100%
switch(config)#
```

This example for aes128cmac shows how to configure the device to synchronize only to time sources that provide authentication key 12 in their NTP packets:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp authentication-key 12 aes128cmac password 0/6/7
switch(config)# ntp server 192.0.2.105 key 12
switch(config)# ntp trusted-key 12
```

```
switch(config)# ntp authenticate
switch(config)# copy running-config startup-config
[#####] 100%
switch(config)#
```

This example shows an NTP access group configuration with the following restrictions:

- Peer restrictions are applied to IP addresses that pass the criteria of the access list named “peer-acl.”
- Serve restrictions are applied to IP addresses that pass the criteria of the access list named “serve-acl.”
- Serve-only restrictions are applied to IP addresses that pass the criteria of the access list named “serve-only-acl.”
- Query-only restrictions are applied to IP addresses that pass the criteria of the access list named “query-only-acl.”

```
switch# configure terminal
switch(config)# ntp peer 10.1.1.1
switch(config)# ntp peer 10.2.2.2
switch(config)# ntp peer 10.3.3.3
switch(config)# ntp peer 10.4.4.4
switch(config)# ntp peer 10.5.5.5
switch(config)# ntp peer 10.6.6.6
switch(config)# ntp peer 10.7.7.7
switch(config)# ntp peer 10.8.8.8
switch(config)# ntp access-group peer peer-acl
switch(config)# ntp access-group serve serve-acl
switch(config)# ntp access-group serve-only serve-only-acl
switch(config)# ntp access-group query-only query-only-acl
switch(config)# ip access-list peer-acl
switch(config-acl)# 10 permit ip host 10.1.1.1 any
switch(config-acl)# 20 permit ip host 10.8.8.8 any
switch(config)# ip access-list serve-acl
switch(config-acl)# 10 permit ip host 10.4.4.4 any
switch(config-acl)# 20 permit ip host 10.5.5.5 any
switch(config)# ip access-list serve-only-acl
switch(config-acl)# 10 permit ip host 10.6.6.6 any
switch(config-acl)# 20 permit ip host 10.7.7.7 any
switch(config)# ip access-list query-only-acl
switch(config-acl)# 10 permit ip host 10.2.2.2 any
switch(config-acl)# 20 permit ip host 10.3.3.3 any
```



- Note** When only a single ACL group is applied, then all the packets relevant for other ACL categories are denied and only packets relevant for the configured ACL group is processed, as mentioned in below scenarios:
- If serve ACL is configured, then only client, control, and private packets are processed and all the other packets are denied.
 - If serve-only ACL is configured, then only client packets are processed and all the other packets are denied.

If more than a single ACL is configured, it follows the order of processing as mentioned in below scenario:

- If serve and serve-only both are configured for the same IP address without match-all configured, where the IP is permitted in serve-acl and denied in serve-only, the client, control, private packets are permitted for that IP.

Additional References

Related Documents

Related Topic	Document Title
Clock manager	Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide

MIBs

MIBs	MIBs Link
MIBs related to NTP	To locate and download supported MIBs, go to the following https://cisco.github.io/cisco-mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html



CHAPTER 10

Configuring CDP

This chapter describes how to configure the Cisco Discovery Protocol (CDP) on Cisco NX-OS devices.

This chapter includes the following sections:

- [About CDP, on page 147](#)
- [Guidelines and Limitations for CDP, on page 148](#)
- [Default Settings for CDP, on page 149](#)
- [Configuring CDP, on page 149](#)
- [Verifying the CDP Configuration, on page 151](#)
- [Configuration Example for CDP, on page 152](#)

About CDP

The Cisco Discovery Protocol (CDP) is a media-independent and protocol-independent protocol that runs on all Cisco-manufactured equipment including routers, bridges, access and communication servers, and switches. You can use CDP to discover and view information about all the Cisco devices that are directly attached to the device.

CDP gathers protocol addresses of neighboring devices and discovers the platform of those devices. CDP runs over the data link layer only. Two systems that support different Layer 3 protocols can learn about each other.

Each device that you configure for CDP sends periodic advertisements to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain hold-time information, which indicates the length of time that a receiving device should hold CDP information before removing it. You can configure the advertisement or refresh timer and the hold timer.

CDP Version-2 (CDPv2) allows you to track instances where the native VLAN ID or port duplex states do not match between connecting devices.

CDP advertises the following type-length-value fields (TLVs):

- Device ID
- Address
- Port ID
- Capabilities
- Version

- Platform
- Native VLAN
- Full or Half Duplex
- MTU
- SysName
- SysObjectID
- Management Address
- Physical Location
- VTP

All CDP packets include a VLAN ID. If you configure CDP on a Layer 2 access port, the CDP packets sent from that access port include the access port VLAN ID. If you configure CDP on a Layer 2 trunk port, the CDP packets sent from that trunk port include the lowest configured VLAN ID allowed on that trunk port. The trunk port can receive CDP packets that include any VLAN ID in the allowed VLAN list for that trunk port. For more information on VLANs, see the [Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide](#).

VTP Feature Support

CDP sends the VLAN Trunking Protocol (VTP) type-length-value field (TLV) if the following conditions are met:

- CDP Version 2 is enabled.
- The VTP feature is enabled.
- A VTP domain name is configured.

You can view the VTP information with the **show cdp neighbors detail** command.

High Availability

Cisco NX-OS supports both stateful and stateless restarts and switchover for CDP. For more information on high availability, see the *Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*.

Virtualization Support

Cisco NX-OS supports one instance of CDP.

Guidelines and Limitations for CDP

CDP has the following configuration guidelines and limitations:

- CDP can discover up to 256 neighbors per port if the port is connected to a hub with 256 connections.

- CDP must be enabled on the device or you cannot enable it on any interfaces.
- You can configure CDP on physical interfaces and port channels only.
- Beginning with Cisco NX-OS Release 10.3(1)F, CDP is supported on the Cisco Nexus 9808 platform switches.
- Beginning with Cisco NX-OS Release 10.4(2)F, CDP is supported on Cisco Nexus 9232E-B1 platform switches.

Default Settings for CDP

This table lists the default settings for CDP parameters.

Parameters	Default
CDP	Enabled globally and on all interfaces
CDP version	Version 2
CDP device ID	Serial number
CDP timer	60 seconds
CDP hold timer	180 seconds

Configuring CDP



Note The Cisco NX-OS commands for this feature may differ from those commands that are used in Cisco IOS.

Enabling or Disabling CDP Globally

CDP is enabled by default. You can disable CDP and then reenabling it.

You must enable CDP on the device before you enable CDP on any interfaces. If CDP is disabled globally and you enable CDP on specified interfaces, CDP will not be active on those interfaces; the system does not return an error message.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.

	Command or Action	Purpose
Step 2	[no] cdp enable Example: switch(config)# cdp enable	Enables or disables the CDP feature on the entire device. It is enabled by default.
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling or Disabling CDP on an Interface

CDP is enabled by default on an interface. You can disable CDP on an interface.

If CDP is disabled globally and you enable CDP on specified interfaces, CDP will not be active on those interfaces; the system does not return an error message.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface interface slot/port Example: switch(config)# interface ethernet 1/2 switch(config-if)#	Enters interface configuration mode.
Step 3	[no] cdp enable Example: switch(config-if)# cdp enable	Enables or disables CDP on this interface. It is enabled by default. Note Make sure that CDP is enabled globally on the device.
Step 4	(Optional) show cdp interface interface slot/port Example: switch(config-if)# show cdp interface ethernet 1/2	Displays CDP information for an interface.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring Optional CDP Parameters

You can use the optional commands in this procedure to modify CDP.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	(Optional) cdp advertise {v1 v2} Example: switch(config)# cdp advertise v1	Sets the CDP version that is supported by the device. The default is v2.
Step 3	(Optional) cdp format device-id {mac-address serial-number system-name} Example: switch(config)# cdp format device-id mac-address	Sets the CDP device ID. The options are as follows: • mac-address—The MAC address of the chassis. • serial-number—The chassis serial number/Organizationally Unique Identifier (OUI). • system-name—The system name or fully qualified domain name. The default is system-name.
Step 4	(Optional) cdp holdtime seconds Example: switch(config)# cdp holdtime 150	Sets the time that CDP holds onto neighbor information before removing it. The range is from 10 to 255 seconds. The default is 180 seconds.
Step 5	(Optional) cdp timer seconds Example: switch(config)# cdp timer 50	Sets the refresh time when CDP sends advertisements to neighbors. The range is from 5 to 254 seconds. The default is 60 seconds.
Step 6	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verifying the CDP Configuration

To display the CDP configuration, perform one of the following tasks:

Command	Purpose
show cdp all	Displays all interfaces that have CDP enabled.
show cdp entry {all name <i>entry-name</i> }	Displays the CDP database entries.
show cdp global	Displays the CDP global parameters.
show cdp interface <i>interface slot/port</i>	Displays the CDP interface status.
show cdp neighbors {device-id interface <i>interface slot/port</i> } [detail]	Displays the CDP neighbor status.
show cdp interface <i>interface slot/port</i>	Displays the CDP traffic statistics on an interface.

Use the **clear cdp counters** command to clear CDP statistics on an interface.

Use the **clear cdp table** command to clear the CDP cache for one or all interfaces.

It is recommended to use the **show cdp neighbors detail** command instead of **show cdp neighbors** command. The **show cdp neighbors** command can display only 13 characters of a platform name. To get the full platform name in the display, use **show cdp neighbors detail** command.

Configuration Example for CDP

This example shows how to enable the CDP feature and configure the refresh and hold timers:

```
configure terminal
cdp enable
cdp timer 50
cdp holdtime 100
```



CHAPTER 11

Configuring System Message Logging

This chapter describes how to configure system message logging on Cisco NX-OS devices.

This chapter contains the following sections:

- [About System Message Logging, on page 153](#)
- [Guidelines and Limitations for System Message Logging, on page 154](#)
- [Default Settings for System Message Logging, on page 155](#)
- [Configuring System Message Logging, on page 156](#)
- [Verifying the System Message Logging Configuration, on page 171](#)
- [Repeated System Logging Messages, on page 171](#)
- [Configuration Example for System Message Logging, on page 172](#)
- [Additional References, on page 172](#)

About System Message Logging

You can use system message logging to control the destination and to filter the severity level of messages that system processes generate. You can configure logging to terminal sessions, a log file, and syslog servers on remote systems.

For more information about the system message format and the messages that the device generates, see the [Cisco NX-OS System Messages Reference](#).

By default, the device outputs messages to terminal sessions and logs system messages to a log file.

The following table describes the severity levels used in system messages. When you configure the severity level, the system outputs messages at that level and lower.

Table 8: System Message Severity Levels

Level	Description
0 – emergency	System unusable
1 – alert	Immediate action needed
2 – critical	Critical condition
3 – error	Error condition

Level	Description
4 – warning	Warning condition
5 – notification	Normal but significant condition
6 – informational	Informational message only
7 – debugging	Appears during debugging only

The device logs the most recent 100 messages of severity 0, 1, or 2 to the NVRAM log. You cannot configure logging to the NVRAM.

You can configure which system messages should be logged based on the facility that generated the message and its severity level.

Syslog Servers

The syslog servers run on remote systems that log system messages based on the syslog protocol. You can configure up to eight IPv4 or IPv6 syslog servers.

To support the same configuration of syslog servers on all switches in a fabric, you can use Cisco Fabric Services (CFS) to distribute the syslog server configuration.



Note When the device first initializes, messages are sent to syslog servers only after the network is initialized.

Secure Syslog Servers

Beginning with Cisco NX-OS Release 9.2(1), you can configure the syslog server with support for a secure TLS transport connectivity to remote logging servers. Additionally, you can enforce the NX-OS switches (client) identity via the mutual authentication configuration. For NX-OS switches, this feature supports TLSv1.1 and TLSv1.2.

The Secure syslog server feature uses the TCP/TLS transport and security protocols to provide device authentication and encryption. This feature enables a Cisco NX-OS device (acting as a client) to make a secure, encrypted outbound connection to remote syslog servers (acting as a server) supporting secure connectivity for logging. With authentication and encryption, this feature allows for a secure communication over an insecure network.

Guidelines and Limitations for System Message Logging

System message logging has the following configuration guidelines and limitations:

- System messages are logged to the console and the log file by default.
- Any system messages that are printed before the syslog server is reachable (such as supervisor active or online messages) cannot be sent to the syslog server.

- Cisco recommends maintaining the logging levels for all processes at default. Increasing the levels to higher values can result in seeing syslog messages that are not intended for customers, can generate false alarms, and are generally supposed to be used for short-term troubleshooting purposes by TAC. Cisco does not provide support for syslog messages at levels above default.
- Due to limitations in Syslog, securePOAP pem file name characters length is limited to 230 characters, though secure POAP supports 256 characters length for a pem file name.
- Beginning with Cisco NX-OS Release 9.2(1), you can configure the syslog server with support for a secure TLS transport connectivity to remote logging servers. This feature supports TLS v1.1 and TLS v1.2.
- Beginning with Cisco NX-OS Release 10.2(4)M, TLS v1.3 is supported for syslog on Cisco Nexus 9000 series platform switches.
- For the secure syslog server(s) to be reachable over an in-band (nonmanagement) interface, the CoPP profile may need tweaks. Especially when multiple logging servers are configured and when many syslogs are generated in a short time (such as, boot up and config application).
- This guideline applies to the user-defined persistent logging file:

The syslog command, **logging logfile**, allows the configuration of the logfile both in persistent (/logflash/log) and non-persistent locations (/log).

The default logfile is named “messages” and this file, along with backup files (if present) messages.1, messages.2, messages.3, messages.4 cannot be deleted, even by the **delete /log/** or **delete logflash:/log/** commands.

There is a provision to configure custom-named logfiles (**logging logfile file-name severity**), however this custom-named file can be deleted by the delete operation. If this occurs, syslog logging does not function.

For example, the custom-named logfile is configured and the same file gets deleted via delete operation. Because this is an intentional delete operation, in order to log the syslog messages on the custom logfiles, you must reconfigure the custom logfile using command **logging logfile file-name severity**. Until this configuration is performed, the syslog logging cannot occur.

- Generally, the syslogs display the local time zone. However, few components such as NGINX display the logs in UTC time zone.
- Beginning with Cisco NX-OS Release 10.3(4a)M, the existing **logging rfc-strict 5424** command (optional) that enables the syslog protocol RFC 5424 is enhanced by adding a new keyword (**full**) as follows:

logging rfc-strict 5424 full

The addition of this keyword ensures complete compliance with the RFC 5424 standard for Syslog Protocol. However, if the values are not available for the [APP-NAME] [PROCID] [MSG-ID] [STRUCTURED-DATA] fields, then the nil value is indicated by a dash (-).

Default Settings for System Message Logging

The following table lists the default settings for the system message logging parameters.

Table 9: Default System Message Logging Parameters

Parameters	Default
Console logging	Enabled at severity level 2
Monitor logging	Enabled at severity level 5
Log file logging	Enabled to log messages at severity level 5
Module logging	Enabled at severity level 5
Facility logging	Enabled
Time-stamp units	Seconds
Syslog server logging	Disabled
Syslog server configuration distribution	Disabled

Configuring System Message Logging



Note Be aware that the Cisco NX-OS commands for this feature might differ from those commands used in Cisco IOS.

Configuring System Message Logging to Terminal Sessions

You can configure the device to log messages by their severity level to console, Telnet, and SSH sessions.

By default, logging is enabled for terminal sessions.



Note The current critical (default) logging level is maintained if the console baud speed is 9600 baud (default). All attempts to change the console logging level will generate an error message. To increase the logging level (above critical), you must change the console baud speed to 38400 baud.

Procedure

	Command or Action	Purpose
Step 1	terminal monitor Example: switch# terminal monitor	Enables the device to log messages to the console.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 3	[no] logging console [severity-level] Example: <pre>switch(config)# logging console 3</pre>	Configures the device to log messages to the console session based on a specified severity level or higher. A lower number indicates a higher severity level. Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging If the severity level is not specified, the default of 2 is used. The no option disables the device's ability to log messages to the console.
Step 4	(Optional) show logging console Example: <pre>switch(config)# show logging console</pre>	Displays the console logging configuration.
Step 5	[no] logging monitor [severity-level] Example: <pre>switch(config)# logging monitor 3</pre>	Enables the device to log messages to the monitor based on a specified severity level or higher. A lower number indicates a higher severity level. Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational

	Command or Action	Purpose
		• 7 – debugging The configuration applies to Telnet and SSH sessions. If the severity level is not specified, the default of 2 is used. The no option disables the device's ability to log messages to the Telnet and SSH sessions.
Step 6	(Optional) show logging monitor Example: <pre>switch(config)# show logging monitor</pre>	Displays the monitor logging configuration.
Step 7	[no] logging message interface type ethernet description Example: <pre>switch(config)# logging message interface type ethernet description</pre>	Enables you to add the description for physical Ethernet interfaces and subinterfaces in the system message log. The description is the same description that was configured on the interface. The no option disables the printing of the interface description in the system message log for physical Ethernet interfaces.
Step 8	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring the Origin ID for Syslog Messages

You can configure Cisco NX-OS to append the hostname, an IP address, or a text string to syslog messages that are sent to remote syslog servers.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Required: logging origin-id {hostname ip ip-address string text-string} Example: <pre>switch(config)# logging origin-id string n9k-switch-abc</pre>	Specifies the hostname, IP address, or text string to be appended to syslog messages that are sent to remote syslog servers.

	Command or Action	Purpose
Step 3	(Optional) show logging origin-id Example: <pre>switch(config)# show logging origin-id Logging origin_id : enabled (string: n9k-switch-abc)</pre>	Displays the configured hostname, IP address, or text string that is appended to syslog messages that are sent to remote syslog servers.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Logging System Messages to a File

You can configure the device to log system messages to a file. By default, system messages are logged to the file `/logflash/log/logfilename`.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] logging logfile logfile-name severity-level [persistent threshold percent size bytes] Example: <pre>switch(config)# logging logfile my_log 6 switch(config)# logging logfile my_log 6 persistent threshold 90</pre>	Configures the nonpersistent or persistent log file parameters. <i>logfile-name</i> : Configures the name of the log file that is used to store system messages. Default filename is "message". <i>severity-level</i> : Configures the minimum severity level to log. A lower number indicates a higher severity level. Default is 5. Range is from 0 through 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational

	Command or Action	Purpose
		• 7 – debugging persistent threshold <i>percent</i>: Optionally configure the threshold percentage for the persistent log file. Range is from 0 through 99. Note Setting persistent threshold to 0 (zero) disables the persistent threshold feature and generates no threshold syslogs. <i>percent</i> configures the percent threshold size of the persistent file. Once the threshold size is reached, an alert notification message is logged. On reaching 100% utilization of the persistent log file, the system sends another syslog message notification. The system then creates a backup file of the existing log file and starts writing into a new log file with the configured threshold percentage applied. In total, the last five backup files are present at most. After five files, the system deletes files based on the oldest modified. Note Persistent logging is a system-enabled feature. Log files are located here: /logflash/log/[filename]. Outputs of the following show commands support the persistent log file feature: • show logging info • show logging The outputs include the following persistent logging information: <pre>Logging logflash: enabled (Severity: notifications) (threshold percentage: 99) Logging logfile: enabled Name - messages: Severity - notifications Size - 4194304</pre> size bytes: Optionally specify maximum file size. Range is from 4096 through 4194304 bytes.
Step 3	logging event {link-status trunk-status} {enable default} Example:	Logs interface events. • link-status—Logs all UP/DOWN and CHANGE messages.

	Command or Action	Purpose
	<pre>switch(config)# logging event link-status default</pre>	• trunk-status—Logs all TRUNK status messages. • enable—Specifies to enable logging to override the port level configuration. • default—Specifies that the default logging configuration is used by interfaces that are not explicitly configured.
Step 4	(Optional) show logging info Example: <pre>switch(config)# show logging info</pre>	Displays the logging configuration.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring Module and Facility Messages Logging

You can configure the severity level and time-stamp units of messages logged by modules and facilities.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] logging module [severity-level] Example: <pre>switch(config)# logging module 3</pre>	Enables module log messages that have the specified severity level or higher. Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging

	Command or Action	Purpose
		If the severity level is not specified, the default of 5 is used. The no option disables module log messages.
Step 3	(Optional) show logging module Example: <code>switch(config)# show logging module</code>	Displays the module logging configuration.
Step 4	[no] logging level <i>facility severity-level</i> Example: <code>switch(config)# logging level aaa 2</code>	Enables logging messages from the specified facility that have the specified severity level or higher. Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging To apply the same severity level to all facilities, use the all facility. For defaults, see the show logging level command. The no option resets the logging severity level for the specified facility to its default level. If you do not specify a facility and severity level, the device resets all facilities to their default levels.
Step 5	(Optional) show logging level <i>[facility]</i> Example: <code>switch(config)# show logging level aaa</code>	Displays the logging level configuration and the system default level by facility. If you do not specify a facility, the device displays levels for all facilities.
Step 6	(Optional) [no] logging level <i>ethpm</i> Example: <code>switch(config)# logging level ethpm ?</code> <code><0-7></code> <code>0-emg,1-alrt,2-crit,3-err,4-war,5-notif,6-infor,7-debug</code> <code>link-down</code> Configure logging level for link down syslog messages	Enables logging of the Ethernet Port Manager link-up/link-down syslog messages at level 3. Use the no option to use the default logging level for Ethernet Port Manager syslog messages.

	Command or Action	Purpose
	<pre> link-up Configure logging level for link up syslog messages switch(config)#logging level ethpm link-down ? error ERRORS notif NOTICE (config)# logging level ethpm link-down error ? <CR> (config)# logging level ethpm link-down notif ? <CR> switch(config)#logging level ethpm link-up ? error ERRORS notif NOTICE (config)# logging level ethpm link-up error ? <CR> (config)# logging level ethpm link-up notif ? <CR> </pre>	
Step 7	[no] logging timestamp {microseconds milliseconds seconds} Example: <pre>switch(config)# logging timestamp milliseconds</pre>	Sets the logging time-stamp units. By default, the units are seconds. Note This command applies to logs that are kept in the switch. It does not apply to the external logging server.
Step 8	(Optional) show logging timestamp Example: <pre>switch(config)# show logging timestamp</pre>	Displays the logging time-stamp units configured.
Step 9	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring Logging Syslogs Compliant to RFC 5424

The command can be modified in the following ways :

- **[no] logging rfc-strict 5424**
- **show logging rfc-strict 5424**

Procedure

	Command or Action	Purpose
Step 1	<code>switch(config)#[no] logging rfc-strict 5424</code>	(optional) Negate a command or set its defaults
Step 2	<code>switch(config)# logging rfc-strict 5424</code>	Modify message logging facilities and set RFC to which messages should be compliant.
Step 3	<code>switch(config)#show logging rfc-strict 5424</code>	Displays the syslogs which will be compliant to RFC 5424

Configuring Syslog Servers



Note Cisco recommends that you configure the syslog server to use the management virtual routing and forwarding (VRF) instance. For more information on VRFs, see Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide.

You can configure up to eight syslog servers that reference remote systems where you want to log system messages.



Note Until Cisco NX-OS Release 10.3(2)F, when the user input certain default values, the running-config of logging server commands showed those default values randomly or inconsistently. However, beginning with Cisco NX-OS Release 10.3(2)F, the running config consistently shows only the non-default values.

For example, in earlier releases, for a certain user input, if the running-config showed `logging server 1.1.1.1 port 514 facility local7 use-vrf` default values, from Cisco NX-OS Release 10.3(2)F onwards, for the same input, the running-config shows only `logging server 1.1.1.1` value. Notice that the default value such as the default port, default facility (local7), and the default VRF are not shown in the running-config.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	<code>[no] logging server host [severity-level [use-vrf vrf-name]]</code> Example: <code>switch(config)# logging server 192.0.2.253</code>	Configures a syslog server at the specified hostname, IPv4, or IPv6 address. You can specify logging of messages to a particular syslog server in a VRF by using the use-vrf keyword. The use-vrf vrf-name keyword

	Command or Action	Purpose
	Example: <pre>switch(config)# logging server 2001::3 5 use-vrf red</pre>	identifies the default or management values for the VRF name. The default VRF is the management VRF, by default. However, the show-running command will not list the default VRF. Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging The default outgoing facility is local7. The no option removes the logging server for the specified host. The first example forwards all messages on facility local 7. The second example forwards messages with severity level 5 or lower to the specified IPv6 address in VRF red. Note After configuring this command, any one of the following server status is displayed: • Configured – Configuration is successful. • No errors found - If the syslog is transmitted to the remote syslog server successfully, this status is displayed. • Temporarily unreachable - If there is a problem with transmission, this status is displayed. However, internally, the system probes the problem with transmission. After a while, when the issue is resolved, the status changes to No errors found.
Step 3	Required: logging source-interface loopback virtual-interface Example: <pre>switch(config)# logging source-interface loopback 5</pre>	Enables a source interface for the remote syslog server. The range for the <i>virtual-interface</i> argument is from 0 to 1023.

	Command or Action	Purpose
Step 4	(Optional) show logging server Example: switch(config)# show logging server	Displays the syslog server configuration.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring Secure Syslog Servers

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] logging server host [severity-level [port port-number]][secure[trustpoint client-identity trustpoint-name]][use-vrf vrf-name]] Example: switch(config)# logging server 192.0.2.253 secure Example: switch(config)# logging server 2001::3 5 secure trustpoint client-identity myCA use-vrf red	Configures a syslog server at the specified hostname or IPv4 or IPv6 address. Optionally, you can enforce a mutual authentication by installing the client identity certificate that is signed by any CA and using the trustpoint client-identity option. The default destination port for a secure TLS connection is 6514.
Step 3	(Optional) logging source-interface interface name Example: switch(config)# logging source-interface lo0	Enables a source interface for the remote syslog server.
Step 4	(Optional) show logging server Example: switch(config)# show logging server	Displays the syslog server configuration. If the secure option is configured, the output will have an entry with the transport information. By default, the transport is UDP if the secure option is not configured.

	Command or Action	Purpose
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the CA Certificate

For the secure syslog feature support, the remote servers must be authenticated via a trustpoint configuration.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] crypto ca trustpoint <i>trustpoint-name</i> Example: switch(config)# crypto ca trustpoint winca switch(config-trustpoint)#	Configures a trustpoint. Note You must configure the ip domain-name before the trustpoint configuration.
Step 3	Required: crypto ca authenticate <i>trustpoint-name</i> Example: switch(config-trustpoint)# crypto ca authenticate winca	Configures a CA certificate for the trustpoint.
Step 4	(Optional) show crypto ca certificate Example: switch(config)# show crypto ca certificates	Displays the configured certificate/chain and the associated trustpoint.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration so that the trustpoint is persistent across the reload of the device.

Enrolling the CA Certificate

For mutual authentication, where the remote server wants the NX-OS switch (the client) to identify, that the peer authentication is mandatory, this is an additional configuration to enroll the certificate on the switch.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	Required: crypto key generate rsa label key name exportable modules 2048 Example: switch(config-trustpoint)# crypto key generate rsa label myKey exportable modulus 2048	Configure an RSA key pair. By default, the Cisco NX-OS software generates an RSA key using 1024 bits.
Step 3	[no] crypto ca trustpoint trustpoint-name Example: switch(config)# crypto ca trustpoint myCA switch(config-trustpoint)#	Configures a trustpoint. Note You must configure the ip domain-name before the trustpoint configuration.
Step 4	Required: rsa keypair key-name Example: switch(config-trustpoint)# rsa keypair myKey	Associates the keypair generated to the trustpoint CA.
Step 5	crypto ca trustpoint trustpoint-name Example: switch(config)# crypto ca authenticate myCA	Configures a CA certificate for the trustpoint.
Step 6	[no] crypto ca enroll trustpoint-name Example: switch(config)# crypto ca enroll myCA	Generate an identity certificate of the switch to enroll it to a CA.
Step 7	crypto ca import trustpoint-name certificate Example: switch(config-trustpoint)# crypto ca import myCA certificate	Imports the identity certificate signed by the CA to the switch.
Step 8	(Optional) show crypto ca certificates Example: switch# show crypto ca certificates	Displays the configured certificate or chain and the associated trustpoint.
Step 9	Required: copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring Syslog Servers on a UNIX or Linux System

You can configure a syslog server on a UNIX or Linux system by adding the following line to the `/etc/syslog.conf` file:

```
facility.level <five tab characters> action
```

The following table describes the syslog fields that you can configure.

Table 10: Syslog fields in `syslog.conf`

Field	Description
Facility	Creator of the message, which can be auth, authpriv, cron, daemon, kern, lpr, mail, mark, news, syslog, user, local0 through local7, or an asterisk (*) for all. These facility designators allow you to control the destination of messages based on their origin. Note Check your configuration before using a local facility.
Level	Minimum severity level at which messages are logged, which can be debug, info, notice, warning, err, crit, alert, emerg, or an asterisk (*) for all. You can use none to disable a facility.
Action	Destination for messages, which can be a filename, a hostname preceded by the at sign (@), a comma-separated list of users, or an asterisk (*) for all logged-in users.

Procedure

Step 1 Log debug messages with the local7 facility in the file `/var/log/myfile.log` by adding the following line to the `/etc/syslog.conf` file:

Example:

```
debug.local7 var/log/myfile.log
```

Step 2 Create the log file by entering these commands at the shell prompt:

Example:

```
$ touch /var/log/myfile.log
$ chmod 666 /var/log/myfile.log
```

Step 3 Make sure the system message logging daemon reads the new changes by checking `myfile.log` after entering this command:

Example:

```
$ kill -HUP ~cat /etc/syslog.pid~
```

Displaying and Clearing Log Files

You can display or clear messages in the log file and the NVRAM.

Procedure

	Command or Action	Purpose
Step 1	Required: show logging last <i>number-lines</i> Example: switch# show logging last 40	Displays the last number of lines in the logging file. You can specify from 1 to 9999 for the last number of lines.
Step 2	show logging logfile duration <i>hh:mm:ss</i> Example: switch# show logging logfile duration 15:10:0	Displays the messages in the log file that have occurred within the duration entered.
Step 3	show logging logfile last-index Example: switch# show logging logfile last-index	Displays the sequence number of the last message in the log file.
Step 4	show logging logfile [start-time <i>yyyy mmm dd hh:mm:ss</i>] [end-time <i>yyyy mmm dd hh:mm:ss</i>] Example: switch# show logging logfile start-time 2013 oct 1 15:10:0	Displays the messages in the log file that have a timestamp within the span entered. If you do not enter an end time, the current time is used. You enter three characters for the month time field and digits for the year and day time fields.
Step 5	show logging logfile [start-seqn <i>number</i>] [end-seqn <i>number</i>] Example: switch# show logging logfile start-seqn 100 end-seqn 400	Displays messages occurring within a range of sequence numbers. If you do not include an end sequence number, the system displays messages from the start number to the last message in the log file.
Step 6	show logging nvram [last <i>number-lines</i>] Example: switch# show logging nvram last 10	Displays the messages in the NVRAM. To limit the number of lines displayed, you can enter the last number of lines to display. You can specify from 1 to 100 for the last number of lines.
Step 7	clear logging logfile [persistent] Example: switch# clear logging logfile	Clears the contents of the log file. persistent: Clears the contents of the log file from the persistent location.
Step 8	clear logging nvram Example:	Clears the logged messages in NVRAM.

	Command or Action	Purpose
	switch# clear logging nvram	

Verifying the System Message Logging Configuration

To display system message logging configuration information, perform one of the following tasks:

Command	Purpose
show logging console	Displays the console logging configuration.
show logging info	Displays the logging configuration.
show logging last <i>number-lines</i>	Displays the last number of lines of the log file.
show logging level [<i>facility</i>]	Displays the facility logging severity level configuration.
show logging logfile duration <i>hh:mm:ss</i>	Displays the messages in the log file that have occurred within the duration entered.
show logging logfile last-index	Displays the sequence number of the last message in the log file.
show logging logfile [start-time <i>yyyy mmm dd hh:mm:ss</i>] [end-time <i>yyyy mmm dd hh:mm:ss</i>]	Displays the messages in the log file based on a start and end date/time.
show logging logfile [start-seqn <i>number</i>] [end-seqn <i>number</i>]	Displays messages occurring within a range of sequence numbers. If you do not include an end sequence number, the system displays messages from the start number to the last message in the log file.
show logging module	Displays the module logging configuration.
show logging monitor	Displays the monitor logging configuration.
show logging nvram [last <i>number-lines</i>]	Displays the messages in the NVRAM log.
show logging server	Displays the syslog server configuration.
show logging timestamp	Displays the logging time-stamp units configuration.

Repeated System Logging Messages

System processes generate logging messages. Depending on the filters used to control which severity levels are generated, a large number of messages can be produced with many of them being repeated.

To make it easier to develop scripts to manage the volume of logging messages, and to eliminate repeated messages from “flooding” the output of the **show logging log** command, the following method of logging repeated messages is used.

In the old method, when the same message was repeated, the default was to state the number of times it reoccurred in the message:

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5
2019 Mar 11 13:43:15 Cisco-customer last message repeated 242 times
```

The new method simply appends the repeat count to the end of the repeated message:

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5

2019 Mar 11 13:43:15 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5 (message repeated 242 times)
```

Configuration Example for System Message Logging

This example shows how to configure system message logging:

```
configure terminal
 logging console 3
 logging monitor 3
 logging logfile my_log 6
 logging module 3
 logging level aaa 2
 logging timestamp milliseconds
 logging server 172.28.254.253
 logging server 172.28.254.254 5 facility local3
 copy running-config startup-config
```

Additional References

Related Documents

Related Topic	Document Title
System messages	<i>Cisco NX-OS System Messages Reference</i>



CHAPTER 12

Configuring Smart Call Home

This chapter describes how to configure the Smart Call Home feature of the Cisco NX-OS devices.

This chapter contains the following sections:

- [About Smart Call Home, on page 173](#)
- [Smart Call Home - Concepts, on page 174](#)
- [Prerequisites for Smart Call Home, on page 179](#)
- [Guidelines and Limitations for Smart Call Home, on page 180](#)
- [Default Settings for Smart Call Home, on page 180](#)
- [Configuring Smart Call Home, on page 181](#)
- [Verifying the Smart Call Home Configuration, on page 198](#)
- [Configuration Examples for Smart Call Home, on page 198](#)
- [Additional References, on page 200](#)

About Smart Call Home

Smart Call Home provides an email-based notification for critical system policies. A range of message formats are available for compatibility with pager services, standard email, or XML-based automated parsing applications. You can use this feature to page a network support engineer, email a Network Operations Center, or use Cisco Smart Call Home services to automatically generate a case with the Technical Assistance Center.

Smart Call Home offers the following features:

- Automatic execution and attachment of relevant CLI command output.
- Multiple message format options such as the following:
 - Short Text—Suitable for pagers or printed reports.
 - Full Text—Fully formatted message information suitable for human reading.
 - XML—Machine-readable format that uses Extensible Markup Language (XML) and Adaptive Messaging Language (AML) XML schema definition (XSD). The AML XSD is published on the Cisco.com website. The XML format enables communication with the Technical Assistance Center.
- Multiple concurrent message destinations. You can configure up to 50 email destination addresses for each destination profile.

Smart Call Home - Concepts

This section explains a few concepts related to Smart Call Home.

Destination Profiles

A destination profile includes the following information:

- One or more alert groups—The group of alerts that trigger a specific Smart Call Home message if the alert occurs.
- One or more email destinations—The list of recipients for the Smart Call Home messages generated by alert groups assigned to this destination profile.
- Message format—The format for the Smart Call Home message (short text, full text, or XML).
- Message severity level—The Smart Call Home severity level that the alert must meet before Cisco NX-OS generates a Smart Call Home message to all email addresses in the destination profile. Cisco NX-OS does not generate an alert if the Smart Call Home severity level of the alert is lower than the message severity level set for the destination profile.

You can also configure a destination profile to allow periodic inventory update messages by using the inventory alert group that will send out periodic messages daily, weekly, or monthly.

Cisco NX-OS supports the following predefined destination profiles:

- CiscoTAC-1—Supports the Cisco-TAC alert group in XML message format. This profile is preconfigured with the callhome@cisco.com email contact, maximum message size, and message severity level 0. You can change any of the default information for this profile.
- full-text-destination—Supports the full text message format.
- short-text-destination—Supports the short text message format.

Smart Call Home Alert Groups

An alert group is a predefined subset of Smart Call Home alerts that are supported in all Cisco Nexus devices. Alert groups allow you to select the set of Smart Call Home alerts that you want to send to a predefined or custom destination profile. The device sends Smart Call Home alerts to email destinations in a destination profile only if that Smart Call Home alert belongs to one of the alert groups associated with that destination profile and if the alert has a Smart Call Home message severity at or above the message severity set in the destination profile.

The following table lists the supported alert groups and the default CLI command output included in Smart Call Home messages generated for the alert group.

Table 11: Alert Groups and Executed Commands

Alert Group	Description	Executed Commands
Cisco-TAC	All critical alerts from the other alert groups destined for Smart Call Home.	Execute commands based on the alert group that originates the alert.
Configuration	Periodic events related to configuration.	show module show version
Diagnostic	Events generated by diagnostics.	show diagnostic result module all detail show diagnostic result module <i>number</i> detail show hardware show logging last 200 show module show sprom all show tech-support gold show tech-support ha show tech-support platform show version
EEM	Events generated by EEM.	show diagnostic result module all detail show diagnostic result module <i>number</i> detail show module show tech-support gold show tech-support ha show tech-support platform
Environmental	Events related to power, fan, and environment-sensing elements such as temperature alarms.	show environment show logging last 200 show module show version

Alert Group	Description	Executed Commands
Inventory	Inventory status that is provided whenever a unit is cold booted or when FRUs are inserted or removed. This alert is considered a noncritical event, and the information is used for status and entitlement.	show inventory show license usage show module show sprom all show system uptime show version
License	Events related to licensing and license violations.	show logging last 200
Linecard hardware	Events related to standard or intelligent switching modules.	show diagnostic result module all detail show diagnostic result module <i>number</i> detail show hardware show logging last 200 show module show sprom all show tech-support ethpm show tech-support gold show tech-support ha show tech-support platform show version
Supervisor hardware	Events related to supervisor modules.	show diagnostic result module all detail show hardware show logging last 200 show module show sprom all show tech-support ethpm show tech-support gold show tech-support ha show tech-support platform show version

Alert Group	Description	Executed Commands
Syslog port group	Events generated by the syslog PORT facility.	show license usage show logging last 200
System	Events generated by failure of a software system that is critical to unit operation.	show diagnostic result module all detail show hardware show logging last 200 show module show sprom all show tech-support ethpm show tech-support gold show tech-support ha show tech-support platform
Test	User-generated test message.	show module show version

Smart Call Home maps the syslog severity level to the corresponding Smart Call Home severity level for syslog port group messages.

You can customize predefined alert groups to execute additional CLI **show** commands when specific events occur and send that **show** output with the Smart Call Home message.

You can add **show** commands only to full text and XML destination profiles. Short text destination profiles do not support additional **show** commands because they only allow 128 bytes of text.

Smart Call Home Message Levels

Smart Call Home allows you to filter messages based on their level of urgency. You can associate each predefined or user-defined destination profile with a Smart Call Home threshold from 0 (least urgent) to 9 (most urgent). The default is 0 (all messages are sent).

Syslog severity levels are mapped to the Smart Call Home message level.



Note Smart Call Home and Syslogs use different severity levels (see the following table). Smart Call Home does not change the syslog message level in the message text.

The following table lists each Smart Call Home message level keyword and the corresponding syslog level for the syslog port alert group.

Table 12: Severity and Syslog Level Mapping

Smart Call Home Level	Keyword	Syslog Level	Description
9	Catastrophic	N/A	Network-wide catastrophic failure.
8	Disaster	N/A	Significant network impact.
7	Fatal	Emergency (0)	System is unusable.
6	Critical	Alert (1)	Critical conditions that indicate that immediate attention is needed.
5	Major	Critical (2)	Major conditions.
4	Minor	Error (3)	Minor conditions.
3	Warning	Warning (4)	Warning conditions.
2	Notification	Notice (5)	Basic notification and informational messages. Possibly independently insignificant.
1	Normal	Information (6)	Normal event signifying return to normal state.
0	Debugging	Debug (7)	Debugging messages.

Obtaining Smart Call Home

If you have a service contract directly with Cisco, you can register for the Smart Call Home service. Smart Call Home analyzes Smart Call Home messages and provides background information and recommendations. For known issues, particularly online diagnostics failures, Automatic Service Requests are generated with the Cisco TAC.

Smart Call Home offers the following features:

- Continuous device health monitoring and real-time diagnostic alerts.
- Analysis of Smart Call Home messages and, if needed, Automatic Service Request generation, routed to the correct TAC team, including detailed diagnostic information to speed problem resolution.
- Secure message transport directly from your device, through an HTTP proxy server, or a downloadable Transport Gateway (TG). You can use a TG aggregation point to support multiple devices or in cases where security dictates that your devices may not be connected directly to the Internet.
- Web-based access to Smart Call Home messages and recommendations, inventory, and configuration information for all Smart Call Home devices. This feature provides access to associated field notices, security advisories, and end-of-life information.

You need the following information to register:

- The SMARTnet contract number for your device
- Your email address

- Your Cisco.com ID

Database Merge Guidelines

When you merge two Smart Call Home databases, the following guidelines apply:

- The merged database contains the following information:
 - A superset of all the destination profiles from the merging devices.
 - The destination profile email addresses and alert groups.
 - Other configuration information (for example, message throttling, or periodic inventory) present in the managing device.
- Destination profile names cannot be duplicated within the merging devices—even though the configurations are different, the names cannot be duplicated. If a profile name is duplicated, one of the duplicate profiles must first be deleted or the merger fails.

High Availability

Both stateful and stateless restarts are supported for Smart Call Home.

Virtualization Support

One instance of Smart Call Home is supported. You can register your contact information at the Smart Call Home web site at the following URL: https://supportforums.cisco.com/community/netpro/solutions/smart_services/smartcallhome

You can test Smart Call Home using the **callhome send** and **callhome test** commands.

Smart Call Home is virtual routing and forwarding (VRF) aware. You can configure Smart Call Home to use a particular VRF to reach the Smart Call Home SMTP server.

Prerequisites for Smart Call Home

Smart Call Home has the following prerequisites:

- To send messages to an email address, you must first configure an email server. To send messages using HTTP, you must have access to an HTTPS server and have a valid certificate installed on the Cisco Nexus device.
- Your device must have IP connectivity to an email server or HTTPS server.
- You must first configure the contact name (SNMP server contact), phone, and street address information. This step is required to determine the origin of messages received.
- If you use Smart Call Home, you need an active service contract for the device that you are configuring.

Guidelines and Limitations for Smart Call Home

Smart Call Home has the following configuration guidelines and limitations:

- If there is no IP connectivity or if the interface in the virtual routing and forwarding (VRF) instance to the profile destination is down, the device cannot send Smart Call Home messages.
- Smart Call Home operates with any SMTP server.
- You can configure up to five SMTP servers for Smart Call Home.
- Link up/down syslog messages do not trigger Smart Call Home messages or alert notifications.
- When configuring Smart Call Home commands such as street address, customer ID, and site ID, you must configure each one of these commands as individual command instead of grouping them with semi-colon separator.
- Callhome does not support specifying a source interface using the **ip http source-interface** command.
- Beginning with Cisco NX-OS Release 10.2(3)F, SMTP-AUTH is supported for secure call home mail transfer on Cisco Nexus 9000 Series platform switches.
- Callhome does not support specifying a source interface using the **ip http source-interface** command. Beginning with Cisco NX-OS Release 10.3(2)F, you can configure the **source-interface** command under callhome to specify a source interface.

Default Settings for Smart Call Home

This table lists the default settings for Smart Call Home parameters.

Table 13: Default Smart Call Home Parameters

Parameters	Default
Destination message size for a message sent in full text format	2,500,000
Destination message size for a message sent in XML format	2,500,000
Destination message size for a message sent in short text format	4000
SMTP server port number if no port is specified	25
SMTP server priority if no priority is specified	50
Alert group association with profile	All for full-text-destination and short-text-destination profiles. The cisco-tac alert group for the CiscoTAC-1 destination profile.
Format type	XML
Smart Call Home message level	0 (zero)

Parameters	Default
HTTP proxy server use	Disabled and no proxy server configured
Periodic inventory notification (periodic-inventory notification)	Enabled by default

Configuring Smart Call Home



Note Be aware that the Cisco NX-OS commands may differ from the Cisco IOS commands.

We recommend that you complete the Smart Call Home configuration procedures in the following sequence:

1. [Configuring Contact Information, on page 181](#)
2. [Creating a Destination Profile, on page 183](#)
3. [Associating an Alert Group with a Destination Profile, on page 186](#)
4. (Optional)
[Adding Show Commands to an Alert Group, on page 187](#)
5. [Enabling or Disabling Smart Call Home, on page 193](#)
6. (Optional)
[Configuring a Source Interface, on page 197](#)
7. (Optional)
[Testing the Smart Call Home Configuration, on page 197](#)

Configuring Contact Information

You must configure the email, phone, and street address information for Smart Call Home. You can optionally configure the contract ID, customer ID, site ID, and switch priority information.

You must configure each one of these Smart Call Home commands as individual command instead of grouping them with semi-colon separator.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.

	Command or Action	Purpose
Step 2	snmp-server contact <i>sys-contact</i> Example: <pre>switch(config)# snmp-server contact personname@companyname.com</pre>	Configures the SNMP sysContact.
Step 3	callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre>	Enters Smart Call Home configuration mode.
Step 4	email-contact <i>email-address</i> Example: <pre>switch(config-callhome)# email-contact admin@Mycompany.com</pre>	Configures the email address for the person primarily responsible for the device. The <i>email-address</i> can be up to 255 alphanumeric characters in email address format. Note You can use any valid email address. The address cannot contain spaces.
Step 5	phone-contact <i>international-phone-number</i> Example: <pre>switch(config-callhome)# phone-contact +1-800-123-4567</pre>	Configures the phone number in international phone number format for the person primarily responsible for the device. The <i>international-phone-number</i> can be up to 17 alphanumeric characters and must be in international phone number format. Note The phone number cannot contain spaces. Use the plus (+) prefix before the number.
Step 6	streetaddress <i>address</i> Example: <pre>switch(config-callhome)# streetaddress 123 Anystreet st. Anytown,AnyWhere</pre>	Configures the street address as an alphanumeric string with white spaces for the person primarily responsible for the device. The <i>address</i> can be up to 255 alphanumeric characters. Spaces are accepted.
Step 7	(Optional) contract-id <i>contract-number</i> Example: <pre>switch(config-callhome)# contract-id Contract5678</pre>	Configures the contract number for this device from the service agreement. The <i>contract-number</i> can be up to 255 alphanumeric characters in free format.
Step 8	(Optional) customer-id <i>customer-number</i> Example: <pre>switch(config-callhome)# customer-id Customer123456</pre>	Configures the customer number for this device from the service agreement. The <i>customer-number</i> can be up to 255 alphanumeric characters in free format.
Step 9	(Optional) site-id <i>site-number</i>	Configures the site number for this device.

	Command or Action	Purpose
	Example: <code>switch(config-callhome)# site-id Site1</code>	The <i>site-number</i> can be up to 255 alphanumeric characters in free format.
Step 10	(Optional) switch-priority number Example: <code>switch(config-callhome)# switch-priority 3</code>	Configures the switch priority for this device. The range is from 0 to 7, with 0 being the highest priority and 7 the lowest. The default is 7.
Step 11	commit Example: <code>switch(config-callhome)# commit</code>	Commits the Smart Call Home configuration commands.
Step 12	(Optional) show callhome Example: <code>switch(config-callhome)# show callhome</code>	Displays a summary of the Smart Call Home configuration.
Step 13	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

What to do next

Create a destination profile.

Creating a Destination Profile

You can create a user-defined destination profile and configure its message format.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	callhome Example: <code>switch(config)# callhome</code> <code>switch(config-callhome)#</code>	Enters Smart Call Home configuration mode.
Step 3	destination-profile name Example:	Creates a new destination profile. The name can be any alphanumeric string up to 31 characters.

	Command or Action	Purpose
	<code>switch(config-callhome)# destination-profile Noc101</code>	
Step 4	destination-profile name format {XML full-txt short-txt} Example: <code>switch(config-callhome)# destination-profile Noc101 format full-txt</code>	Sets the message format for the profile. The name can be any alphanumeric string up to 31 characters.
Step 5	commit Example: <code>switch(config-callhome)# commit</code>	Commits the Smart Call Home configuration commands.
Step 6	(Optional) show callhome destination-profile [profile name] Example: <code>switch(config-callhome)# show callhome destination-profile profile Noc101</code>	Displays information about one or more destination profiles.
Step 7	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

What to do next

Associate one or more alert groups with a destination profile.

Modifying a Destination Profile

You can modify the following attributes for a predefined or user-defined destination profile:

- Destination email address—The actual address, pertinent to the transport mechanism, to which the alert should be sent.
- Destination URL—The HTTP or HTTPS URL that defines where alerts should be sent.
- Transport method—The email or HTTP transport that determines which type of destination addresses are used.
- Message formatting—The message format used for sending the alert (full text, short text, or XML).
- Message level—The Smart Call Home message severity level for this destination profile.
- Message size—The allowed length of a Smart Call Home message sent to the email addresses in this destination profile.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre>	Enters Smart Call Home configuration mode.
Step 3	destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> email-addr <i>address</i> Example: <pre>switch(config-callhome)# destination-profile full-txt-destination email-addr person@place.com</pre>	Configures an email address for a user-defined or predefined destination profile. You can configure up to 50 email addresses in a destination profile.
Step 4	destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> http <i>address</i> Example: <pre>switch(config-callhome)# destination-profile CiscoTAC-1 http https://tools.cisco.com/its/service/odbe/services/DOEService</pre>	Configures an HTTP or HTTPS URL for a user-defined or predefined destination profile. The URL can be up to 255 characters.
Step 5	destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> transport-method <i>{email http}</i> Example: <pre>switch(config-callhome)# destination-profile CiscoTAC-1 transport-method http</pre>	Configures an email or HTTP transport method for a user-defined or predefined destination profile. The type of transport method that you choose determines the configured destination addresses of that type.
Step 6	destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> message-level <i>number</i> Example: <pre>switch(config-callhome)# destination-profile full-txt-destination message-level 5</pre>	Configures the Smart Call Home message severity level for this destination profile. Cisco NX-OS sends only alerts that have a matching or higher Smart Call Home severity level to destinations in this profile. The range is from 0 to 9, where 9 is the highest severity level.

	Command or Action	Purpose
Step 7	destination-profile { <i>name</i> CiscoTAC-1 full-txt-destination short-txt-destination } message-size <i>number</i> Example: <pre>switch(config-callhome)# destination-profile full-txt-destination message-size 100000</pre>	Configures the maximum message size for this destination profile. The range is from 0 to 5000000. The default is 2500000.
Step 8	commit Example: <pre>switch(config-callhome)# commit</pre>	Commits the Smart Call Home configuration commands.
Step 9	(Optional) show callhome destination-profile [<i>profile name</i>] Example: <pre>switch(config-callhome)# show callhome destination-profile profile full-text-destination</pre>	Displays information about one or more destination profiles.
Step 10	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

What to do next

Associate one or more alert groups with a destination profile.

Associating an Alert Group with a Destination Profile

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre>	Enters Smart Call Home configuration mode.

	Command or Action	Purpose
Step 3	destination-profile { <i>name</i> CiscoTAC-1 full-txt-destination short-txt-destination } alert-group { All Cisco-TAC Configuration Diagnostic EEM Environmental Inventory License Supervisor-Hardware Syslog-group-port System Test } Example: <pre>switch(config-callhome)# destination-profile Noc101 alert-group All</pre>	Associates an alert group with this destination profile. Use the All keyword to associate all alert groups with the destination profile.
Step 4	commit Example: <pre>switch(config-callhome)# commit</pre>	Commits the Smart Call Home configuration commands.
Step 5	(Optional) show callhome destination-profile [<i>profile name</i>] Example: <pre>switch(config-callhome)# show callhome destination-profile profile Noc101</pre>	Displays information about one or more destination profiles.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

What to do next

Optionally add **show** commands to an alert group and then configure the SMTP email server.

Adding Show Commands to an Alert Group

You can assign a maximum of five user-defined CLI **show** commands to an alert group.



Note You cannot add user-defined CLI **show** commands to the CiscoTAC-1 destination profile.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre>	Enters Smart Call Home configuration mode.
Step 3	alert-group {Configuration Diagnostic EEM Environmental Inventory License Supervisor-Hardware Syslog-group-port System Test} user-def-cmd show-cmd Example: <pre>switch(config-callhome)# alert-group Configuration user-def-cmd show ip route</pre>	Adds the show command output to any Smart Call Home messages sent for this alert group. Only valid show commands are accepted.
Step 4	commit Example: <pre>switch(config-callhome)# commit</pre>	Commits the Smart Call Home configuration commands.
Step 5	(Optional) show callhome user-def-cmds Example: <pre>switch(config-callhome)# show callhome user-def-cmds</pre>	Displays information about all user-defined show commands added to alert groups.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

What to do next

Configure Smart Call Home to connect to the SMTP email server.

Configuring the Email Server

You must configure the SMTP server address for the Smart Call Home functionality to work. You can also configure the from and reply-to email addresses.

You can configure up to five SMTP servers for Smart Call Home. The servers are tried based on their priority. The highest priority server is tried first. If the message fails to be sent, the next server in the list is tried until the limit is exhausted. If two servers have equal priority, the one that was configured earlier is tried first.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	callhome Example: switch(config)# callhome switch(config-callhome)#	Enters Smart Call Home configuration mode.
Step 3	transport email mail-server ip-address [port number] [priority number] [use-vrf vrf-name] Example: switch(config-callhome)# transport email mail-server 192.0.2.1 use-vrf Red	Configures the SMTP server as the domain name server (DNS) name, IPv4 address, or IPv6 address. Optionally configures the port number. The port range is from 1 to 65535. The default port number is 25. Also optionally configures the priority of the SMTP server. The priority range is from 1 to 100, with 1 being the highest priority and 100 the lowest. If you do not specify a priority, the default value of 50 is used. Also optionally configures the VRF to use when communicating with this SMTP server. The VRF specified is not used to send messages using HTTP.
Step 4	(Optional) transport email from email-address Example: switch(config-callhome)# transport email from person@company.com	Configures the email from field for Smart Call Home messages.
Step 5	(Optional) transport email reply-to email-address Example: switch(config-callhome)# transport email reply-to person@company.com	Configures the email reply-to field for Smart Call Home messages.
Step 6	commit Example: switch(config-callhome)# commit	Commits the Smart Call Home configuration commands.
Step 7	(Optional) show callhome transport Example: switch(config-callhome)# show callhome transport	Displays the transport-related configuration for Smart Call Home.
Step 8	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

What to do next

Optionally use VRFs to send Smart Call Home messages over HTTP.

Configuring VRFs To Send Messages Using HTTP

You can use VRFs to send Smart Call Home messages over HTTP. If HTTP VRFs are not configured, the default VRF is used to transport messages over HTTP.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre>	Enters Smart Call Home configuration mode.
Step 3	transport http use-vrf <i>vrf-name</i> Example: <pre>switch(config-callhome)# transport http use-vrf Blue</pre>	Configures the VRF used to send email and other Smart Call Home messages over HTTP.
Step 4	commit Example: <pre>switch(config-callhome)# commit</pre>	Commits the Smart Call Home configuration commands.
Step 5	(Optional) show callhome Example: <pre>switch(config-callhome)# show callhome</pre>	Displays information about Smart Call Home.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

What to do next

Optionally configure Smart Call Home to send HTTP messages through an HTTP proxy server.

Configuring an HTTP Proxy Server

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre>	Enters Smart Call Home configuration mode.
Step 3	transport http proxy server ip-address [port number] Example: <pre>switch(config-callhome)# transport http proxy server 192.0.2.1</pre>	Configures the HTTP proxy server domain name server (DNS) name, IPv4 address, or IPv6 address. Optionally configures the port number. The port range is from 1 to 65535. The default port number is 8080.
Step 4	transport http proxy enable Example: <pre>switch(config-callhome)# transport http proxy enable</pre>	Enables Smart Call Home to send all HTTP messages through the HTTP proxy server. Note You can execute this command only after the proxy server address has been configured. Note The VRF used for transporting messages through the proxy server is the same as that configured using the transport http use-vrf command.
Step 5	commit Example: <pre>switch(config-callhome)# commit</pre>	Commits the Smart Call Home configuration commands.
Step 6	(Optional) show callhome transport Example: <pre>switch(config-callhome)# show callhome transport</pre>	Displays the transport-related configuration for Smart Call Home.
Step 7	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

What to do next

Optionally configure your device to periodically send inventory notifications.

Configuring Periodic Inventory Notifications

You can configure the device to periodically send a message with an inventory of all software services currently enabled and running on the device along with hardware inventory information. The device generates two Smart Call Home notifications: periodic configuration messages and periodic inventory messages.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre>	Enters Smart Call Home configuration mode.
Step 3	periodic-inventory notification [interval days] [timeofday time] Example: <pre>switch(config-callhome)# periodic-inventory notification interval 20</pre>	Configures periodic inventory messages. The interval range is from 1 to 30 days, and the default is 7 days. The <i>time</i> argument is in HH:MM format. It defines at what time of the day every <i>X</i> days an update is sent (where <i>X</i> is the update interval).
Step 4	commit Example: <pre>switch(config-callhome)# commit</pre>	Commits the Smart Call Home configuration commands.
Step 5	(Optional) show callhome Example: <pre>switch(config-callhome)# show callhome</pre>	Displays information about Smart Call Home.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

What to do next

Optionally disable duplicate message throttling.

Disabling Duplicate Message Throttling

You can limit the number of duplicate messages received for the same event. By default, the device limits the number of duplicate messages received for the same event. If the number of duplicate messages sent exceeds 30 messages within a 2-hour time frame, the device discards further messages for that alert type.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	callhome Example: <code>switch(config)# callhome</code> <code>switch(config-callhome)#</code>	Enters Smart Call Home configuration mode.
Step 3	no duplicate-message throttle Example: <code>switch(config-callhome)# no</code> <code>duplicate-message throttle</code>	Disables duplicate message throttling for Smart Call Home. Duplicate message throttling is enabled by default.
Step 4	commit Example: <code>switch(config-callhome)# commit</code>	Commits the Smart Call Home configuration commands.
Step 5	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config</code> <code>startup-config</code>	Copies the running configuration to the startup configuration.

What to do next

Enable Smart Call Home.

Enabling or Disabling Smart Call Home

Once you have configured the contact information, you can enable the Smart Call Home function.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	callhome Example: switch(config)# callhome switch(config-callhome)#	Enters Smart Call Home configuration mode.
Step 3	[no] enable Example: switch(config-callhome)# enable	Enables or disables Smart Call Home. Smart Call Home is disabled by default.
Step 4	commit Example: switch(config-callhome)# commit	Commits the Smart Call Home configuration commands.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

What to do next

Optionally generate a test message.

Configuring SMTP-AUTH for Call Home Mail Transfer

You can use the SMTP-AUTH feature for call home mail transfer to share mails in a secure way using standard SMTP-AUTH TCP port 587 or 465, or any other user-defined port, instead of clear text over port 25. This feature is supported from Cisco NX-OS Release 10.2(3)F.

Before you begin

- SMTP-AUTH server certificate should be installed on the switch.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	callhome Example:	Enters Smart Call Home configuration mode.

	Command or Action	Purpose
	<pre>switch(config)# callhome switch(config-callhome)#</pre>	
Step 3	email-contact <i>email-address</i> Example: <pre>switch(config-callhome)# email-contact admin@Mycompany.com</pre>	Configures the email address for the person primarily responsible for the device. The <i>email-address</i> can be up to 255 alphanumeric characters in email address format. Note You can use any valid email address. The address cannot contain spaces.
Step 4	destination-profile <i>name</i> Example: <pre>switch(config-callhome)# destination-profile testProfile-1</pre>	Creates a new destination profile. The name can be any alphanumeric string up to 31 characters.
Step 5	destination-profile <i>name</i> format {XML full-txt short-txt} Example: <pre>switch(config-callhome)# destination-profile testProfile-1 format XML</pre>	Sets the message format for the profile. The name can be any alphanumeric string up to 31 characters.
Step 6	destination-profile <i>name</i> email-address <i>email-address</i> Example: <pre>switch(config-callhome)# destination-profile testProfile-1 index 1 email address person@company.com</pre>	Configures an email address to which the secure mail must be delivered. You can configure up to 50 email addresses in a destination profile.
Step 7	destination-profile <i>name</i> alert-group all Example: <pre>switch(config-callhome)# destination-profile testProfile-1 alert-group all</pre>	Associates all the alert groups with the destination profile.
Step 8	transport email from <i>callhome_email-address</i> Example: <pre>switch(config)# transport email from callhome_person@company.com</pre>	Configures the email from callhome field for Smart Call Home messages.
Step 9	transport email smtp-server <i>hostname/ip-address</i> port 465 use-vrf <i>vrf-name</i> Example:	transport email smtp-server <i>hostname/ip-address</i> port 587 use-vrf <i>vrf-name</i>

	Command or Action	Purpose
	<pre>switch(config)# transport email smtp-server 10.1.1.174 port 465 use-vrf management switch(config)# transport email smtp-server 10.1.1.174 port 587 use-vrf management</pre>	Enables SMTP-AUTH mail transfer method; STARTTLS-based SMTP-AUTH over the standard TCP ports, that is, 465 and 587 ports.
Step 10	transport email username <i>username</i> passwd <i>password</i> {cleartext encrypted} Example: <pre>switch(config)# transport email username user1 passwd Y2FsbGhvbWUK encrypted</pre>	Accepts username and password and passes these details for SMTP-AUTH authentication. The username should be alphanumeric and must be less than 256 bytes. Password option can be entered in cleartext or encrypted format (if the user already has the encrypted password). The password length must be less than 64 bytes for the cleartext option and less than 256 bytes for the encrypted option. Note SMTP-AUTH fails in the following scenarios: • if the password in cleartext is more than 56 characters in length. • if the password has any of the following special characters: • Dollar sign - \$ • Parentheses - (and) • Ampersand - & • Square Brackets - [and] • Semicolon - ; • Question mark - ? • Vertical bar or pipe - • Apostrophe - ' • Quotation marks - ', ", ' , ' , " , and " • Less-than and More-than signs - > and <
Step 11	(Optional) transport http use-vrf <i>vrf-name</i> Example: <pre>switch(config)# transport http use-vrf management</pre>	Configures the VRF used to send email and other Smart Call Home messages over HTTP.
Step 12	[no] enable	Enables Smart Call Home.

	Command or Action	Purpose
	Example: <code>switch(config)# enable</code>	The no form of this command disables Smart Call Home.

Configuring a Source Interface

Use this procedure to identify the source-interface in a VRF through which the packet is sent.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	callhome Example: <code>switch(config)# callhome</code> <code>switch(config-callhome)#</code>	Enters Smart Call Home configuration mode.
Step 3	source interface <i>interface_name</i> Example: <code>switch(config-callhome)# source-interface</code> <code>Ethernet1/1</code>	Defines the source interface for call home http(s) transfer, and when call home is used as the transport for Smart Licensing.

Testing the Smart Call Home Configuration

You can generate a test message to test your Smart Call Home communications.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	callhome Example: <code>switch(config)# callhome</code> <code>switch(config-callhome)#</code>	Enters Smart Call Home configuration mode.

	Command or Action	Purpose
Step 3	callhome send [configuration diagnostic] Example: switch(config-callhome)# callhome send diagnostic	Sends the specified Smart Call Home test message to all configured destinations.
Step 4	callhome test Example: switch(config-callhome)# callhome test	Sends a test message to all configured destinations.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verifying the Smart Call Home Configuration

To display Smart Call Home configuration information, perform one of the following tasks:

Command	Purpose
show callhome	Displays the Smart Call Home configuration.
show callhome destination-profile name	Displays one or more Smart Call Home destination profiles.
show callhome transport	Displays the transport-related configuration for Smart Call Home.
show callhome user-def-cmds	Displays CLI commands added to any alert groups.
show running-config callhome [all]	Displays the running configuration for Smart Call Home.
show startup-config callhome	Displays the startup configuration for Smart Call Home.
show tech-support callhome	Displays the technical support output for Smart Call Home.

Configuration Examples for Smart Call Home

This example shows how to create a destination profile called Noc101, associate the Configuration alert group to that profile, configure contact and email information, and specify the VRF used to send Smart Call Home messages over HTTP:

```
configure terminal
snmp-server contact person@company.com
callhome
distribute
email-contact admin@Mycompany.com
phone-contact +1-800-123-4567
streetaddress 123 Anystreet st. Anytown,AnyWhere
```

```

destination-profile Noc101 format full-txt
destination-profile full-text-destination email-addr person@company.com
destination-profile full-text-destination message-level 5
destination-profile Noc101 alert-group Configuration
alert-group Configuration user-def-cmd show ip route
transport email mail-server 192.0.2.10 priority 1
transport http use-vrf Blue
enable
commit

```

This example shows how to configure multiple SMTP servers for Smart Call Home messages:

```

configure terminal
callhome
transport email mail-server 192.0.2.10 priority 4
transport email mail-server 172.21.34.193
transport email smtp-server 10.1.1.174
transport email mail-server 64.72.101.213 priority 60
transport email from person@company.com
transport email reply-to person@company.com
commit

```



Note Configuration of multiple smtp-servers for authentication purpose using the **callhome email mail-server** command is not supported.

Based on the configuration above, the SMTP servers would be tried in this order:

10.1.1.174 (priority 0)
 192.0.2.10 (priority 4)
 172.21.34.193 (priority 50, which is the default)
 64.72.101.213 (priority 60)



Note The **transport email smtp-server** command has a priority of 0, which is the highest. The server specified by this command is tried first followed by the servers specified by the **transport email mail-server** commands in order of priority.

This example shows how to configure Smart Call Home to send HTTP messages through an HTTP proxy server:

```

configure terminal
callhome
transport http proxy server 10.10.10.1 port 4
transport http proxy enable
commit

```

This example shows how to configure SMTP-AUTH servers for call home mail transfer:

```

callhome
email-contact admin@Mycompany.com
destination-profile testProfile-1
destination-profile testProfile-1 format XML
destination-profile testProfile-1 index 1 email-addr person@company.com

```

```

destination-profile testProfile-1 alert-group all
destination-profile full_txt alert-group test
transport email from callhome_person@company.com
transport email smtp-server 10.1.1.174 port 587 use-vrf management
transport email username user1 passwd Y2FsbGhvbWUK encrypted
transport http use-vrf management
enable

```

Additional References

Event Triggers

The following table lists the event triggers and their Smart Call Home message severity levels.

Alert Group	Event Name	Description	Smart Call Home Severity Level
Configuration	PERIODIC_CONFIGURATION	Periodic configuration update message.	2
Diagnostic	DIAGNOSTIC_MAJOR_ALERT	GOLD generated a major alert.	7
	DIAGNOSTIC_MINOR_ALERT	GOLD generated a minor alert.	4
	DIAGNOSTIC_NORMAL_ALERT	Smart Call Home generated a normal diagnostic alert.	2
Environmental and CISCO_TAC	FAN_FAILURE	Cooling fan has failed.	5
	POWER_SUPPLY_ALERT	Power supply warning has occurred.	6
	POWER_SUPPLY_FAILURE	Power supply has failed.	6
	POWER_SUPPLY_SHUTDOWN	Power supply has shut down.	6
	TEMPERATURE_ALARM	Thermal sensor going bad.	6
	TEMPERATURE_MAJOR_ALARM	Thermal sensor indicates temperature has reached operating major threshold.	6
	TEMPERATURE_MINOR_ALARM	Thermal sensor indicates temperature has reached operating minor threshold.	4
Inventory and CISCO_TAC	COLD_BOOT	Switch is powered up and reset to a cold boot sequence.	2
	HARDWARE_INSERTION	New piece of hardware has been inserted into the chassis.	2
	HARDWARE_REMOVAL	Hardware has been removed from the chassis.	2
	PERIODIC_INVENTORY	Periodic inventory message has been generated.	2

Alert Group	Event Name	Description	Smart Call Home Severity Level
License	LICENSE_VIOLATION	Feature in use is not licensed and is turned off after grace period expiration.	6
Line module Hardware and CISCO_TAC	LINEmodule_FAILURE	Module operation has failed.	7
Supervisor Hardware and CISCO_TAC	SUP_FAILURE	Supervisor module operation has failed.	7
Syslog-group-port	PORT_FAILURE	syslog message that corresponds to the port facility has been generated.	6
	SYSLOG_ALERT	syslog alert message has been generated. Note Link up/down syslog messages do not trigger Smart Call Home messages or alert notifications.	5
System and CISCO_TAC	SW_CRASH	Software process has failed with a stateless restart, indicating an interruption of a service. Messages are sent for process crashes on supervisor modules.	5
	SW_SYSTEM_INCONSISTENT	Inconsistency has been detected in software or file system.	5
Test and CISCO_TAC	TEST	User generated test has occurred.	2

Message Formats

Smart Call Home supports the following message formats:

Short Text Message Format

The following table describes the short text formatting option for all message types.

Data Item	Description
Device identification	Configured device name
Date/time stamp	Time stamp of the triggering event
Error isolation message	Plain English description of triggering event

Data Item	Description
Alarm urgency level	Error level such as that applied to system message

Common Event Message Fields

The following table describes the first set of common event message fields for full text or XML messages.

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Timestamp	Date and time stamp of event in ISO time notation: YYYY-MM-DD HH:MM:SS GMT+HH:MM.	/aml/header/time
Message name	Name of message.	/aml/header/name
Message type	Name of message type, such as reactive or proactive.	/aml/header/type
Message group	Name of alert group, such as syslog.	/aml/header/group
Severity level	Severity level of message.	/aml/header/level
Source ID	Product type for routing, such as the Cisco Nexus 9000 Series switch.	/aml/header/source
Device ID	Unique device identifier (UDI) for the end device that generated the message. This field should be empty if the message is nonspecific to a device. The format is <i>type@Sid@serial</i> . <i>type</i> is the product model number from the backplane IDPROM. <i>@</i> is a separator character. <i>Sid</i> is C, identifying the serial ID as a chassis serial number. <i>serial</i> is the number identified by the Sid field. An example is N9K-C9508@C@12345678.	/aml/ header/deviceId
Customer ID	Optional user-configurable field used for contract information or other ID by any support service.	/aml/ header/customerID
Contract ID	Optional user-configurable field used for contract information or other ID by any support service.	/aml/ header /contractId

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Site ID	Optional user-configurable field used for Cisco-supplied site ID or other data meaningful to alternate support service.	/aml/ header/siteId
Server ID	If the message is generated from the device, this ID is the unique device identifier (UDI) of the device. The format is <i>type@Sid@serial</i>. • <i>type</i> is the product model number from the backplane IDPROM. • <i>@</i> is a separator character. • <i>Sid</i> is C, identifying the serial ID as a chassis serial number. • <i>serial</i> is the number identified by the Sid field. An example is N9K-C9508@C@12345678.	/aml/header/serverId
Message description	Short text that describes the error.	/aml/body/msgDesc
Device name	Node that experienced the event (hostname of the device).	/aml/body/sysName
Contact name	Name of person to contact for issues associated with the node that experienced the event.	/aml/body/sysContact
Contact email	Email address of person identified as the contact for this unit.	/aml/body/sysContactEmail
Contact phone number	Phone number of the person identified as the contact for this unit.	/aml/body/sysContactPhone Number
Street address	Optional field that contains the street address for RMA part shipments associated with this unit.	/aml/body/sysStreetAddress
Model name	Model name of the device (the specific model as part of a product family name).	/aml/body/chassis/name
Serial number	Chassis serial number of the unit.	/aml/body/chassis/serialNo
Chassis part number	Top assembly number of the chassis.	/aml/body/chassis/partNo

Alert Group Message Fields

The following table describes the fields specific to alert group messages for full text and XML. These fields may be repeated if multiple CLI commands are executed for an alert group.

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Command output name	Exact name of the issued CLI command.	/aml/attachments/attachment/name
Attachment type	Specific command output.	/aml/attachments/attachment/type
MIME type	Either plain text or encoding type.	/aml/attachments/attachment/mime
Command output text	Output of command automatically executed.	/aml/attachments/attachment/atdata

Fields for Reactive and Proactive Event Messages

The following table describes the reactive and proactive event message format for full text or XML messages.

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Chassis hardware version	Hardware version of chassis.	/aml/body/chassis/hwVersion
Supervisor module software version	Top-level software version.	/aml/body/chassis/swVersion
Affected FRU name	Name of the affected FRU that is generating the event message.	/aml/body/fru/name
Affected FRU serial number	Serial number of the affected FRU.	/aml/body/fru/serialNo
Affected FRU part number	Part number of the affected FRU.	/aml/body/fru/partNo
FRU slot	Slot number of the FRU that is generating the event message.	/aml/body/fru/slot
FRU hardware version	Hardware version of the affected FRU.	/aml/body/fru/hwVersion
FRU software version	Software version(s) that is running on the affected FRU.	/aml/body/fru/swVersion

Fields for Inventory Event Messages

The following table describes the inventory event message format for full text or XML messages.

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Chassis hardware version	Hardware version of the chassis.	/aml/body/chassis/hwVersion
Supervisor module software version	Top-level software version.	/aml/body/chassis/swVersion
FRU name	Name of the affected FRU that is generating the event message.	/aml/body/fru/name

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
FRU s/n	Serial number of the FRU.	/aml/body/fru/serialNo
FRU part number	Part number of the FRU.	/aml/body/fru/partNo
FRU slot	Slot number of the FRU.	/aml/body/fru/slot
FRU hardware version	Hardware version of the FRU.	/aml/body/fru/hwVersion
FRU software version	Software version(s) that is running on the FRU.	/aml/body/fru/swVersion

Fields for User-Generated Test Messages

The following table describes the user-generated test message format for full text or XML.

Data Item (Plain Text and XML)	Description (Plain Text and XML)	XML Tag (XML Only)
Process ID	Unique process ID.	/aml/body/process/id
Process state	State of process (for example, running or halted).	/aml/body/process/processState
Process exception	Exception or reason code.	/aml/body/process/exception

Sample Syslog Alert Notification in Full-Text Format

This sample shows the full-text format for a syslog port alert-group notification:

```
Severity Level:5
Series:Nexus9000
Switch Priority:0
Device Id:N9K-C9508C@TXX12345678
Server Id:N9K-C9508C@TXX12345678
Time of Event:2013-05-17 16:31:33 GMT+0000 Message Name:
Message Type:syslog
System Name:dc3-test
Contact Name:Jay Tester
Contact Email:contact@example.com
Contact Phone:+91-80-1234-5678
Street Address:#1 Any Street
Event Description:SYSLOG_ALERT 2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error
(0x20) while communicating with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP
(for:RID_PORT: Ethernet3/1)

syslog_facility:ETHPORT
start chassis information:
Affected Chassis:N9K-C9508
Affected Chassis Serial Number:TXX12345678 Affected Chassis Hardware Version:0.405 Affected
Chassis Software Version:6.1(2) Affected Chassis Part No:11-11111-11 end chassis information:
start attachment
  name:show logging logfile | tail -n 200
  type:text
  data:
    2013 May 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages) cleared
    by user
    2013 May 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
    /dev/ttyS0 /dev/ttyS0_console
    2013 May 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
```

```

/dev/ttyS0 /dev/ttyS0_console
2013 May 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16: Invalid
argument: - sshd[14484]
2013 May 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 12:11:18 dc3-test %SYSMGR-STANDBY-5-SUBPROC_TERMINATED: "System Manager
(gsync controller)" (PID 12000) has finished with error code
SYSMGR_EXITCODE_GSYNCFAILED_NONFATAL (12).
2013 May 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 16:28:44 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 3504) hasn't
caught signal 9 (no core).
2013 May 17 16:29:08 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero.
2013 May 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23210)
hasn't caught signal 9 (no core).
2013 May 17 16:29:17 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero.
2013 May 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23294)
hasn't caught signal 9 (no core).
2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is
becoming active (pre-start phase).
2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is becoming
active.
2013 May 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send failed -
device_test
2013 May 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR: netstack [4336] Unrecognized
message from MRIB. Major type 1807
2013 May 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN
2013 May 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10 - ntpd[19045]

2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined - ntpd[19045]

2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19072]
2013 May 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19073]
2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19079]
2013 May 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP
2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19105]
2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 2 present but
all AC inputs are not connected, ac-redundancy might be affected
2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 3 present but
all AC inputs are not connected, ac-redundancy might be affected

```

```

2013 May 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
2013 May 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2013 May 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2013 May 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2013 May 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2013 May 17 16:30:47 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 4820) hasn't
caught signal 9 (no core).
2013 May 17 16:31:02 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24239)
hasn't caught signal 9 (no core).
2013 May 17 16:31:14 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24401)
hasn't caught signal 9 (no core).
2013 May 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
2013 May 17 16:31:23 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24407)
hasn't caught signal 9 (no core).
2013 May 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2013 May 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2013 May 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2013 May 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR: netstack [4336] (null)
2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1)
end attachment start attachment
type:text
data:

dc3-test interfaces:
Ethernet3/1    Ethernet3/2    Ethernet3/3
Ethernet3/4    Ethernet3/5    Ethernet3/6
Ethernet3/7    Ethernet3/8    Ethernet3/9
Ethernet3/10   Ethernet3/11   Ethernet3/12
Ethernet3/13   Ethernet3/14   Ethernet3/15
Ethernet3/16   Ethernet3/17   Ethernet3/18
Ethernet3/19   Ethernet3/20   Ethernet3/21
Ethernet3/22   Ethernet3/23   Ethernet3/24
Ethernet3/25   Ethernet3/29   Ethernet3/30
Ethernet3/31   Ethernet3/32   Ethernet3/33
Ethernet3/34   Ethernet3/35   Ethernet3/36
Ethernet3/37   Ethernet3/38   Ethernet3/39
Ethernet3/40   Ethernet3/41   Ethernet3/42
Ethernet3/43   Ethernet3/44   Ethernet3/45
Ethernet3/46   Ethernet3/47   Ethernet3/48
end attachment
start attachment
type:text
data:
end attachment
start attachment
name:show license usage
type:text
data:

```

```

Feature Ins Lic Status Expiry Date Comments
Count
-----
LAN_ENTERPRISE_SERVICES_PKG Yes - Unused Never -
-----
end attachment

```

Sample Syslog Alert Notification in XML Format

This sample shows the XML format for a syslog port alert-group notification:

```

<?xml version="1.0" encoding="UTF-8" ?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.cisco.com/2004/01/aml-session"
soap-env:mustUnderstand="true"
soap-env:role="http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.cisco.com/neddce/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.cisco.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.cisco.com/appliance/uri</aml-session:From>
<aml-session:MessageId>1004:TXX12345678:478F82E6</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.cisco.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.cisco.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2013-05-17 16:31:33 GMT+0000</aml-block:CreationDate>
<aml-block:Builder> <aml-block:Name>DC3</aml-block:Name>
<aml-block:Version>4.1</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>1005:TXX12345678:478F82E6</aml-block:GroupId>
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>5</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:CallHome xmlns:ch="http://www.cisco.com/2005/05/callhome" version="1.0">
<ch:EventTime>2013-05-17 16:31:33 GMT+0000</ch:EventTime> <ch:MessageDescription>SYSLOG_ALERT
2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1)
</ch:MessageDescription>
<ch:Event> <ch:Type>syslog</ch:Type> <ch:SubType></ch:SubType> <ch:Brand>Cisco</ch:Brand>
<ch:Series>Nexus9000</ch:Series> </ch:Event> <ch:CustomerData> <ch:UserData>
<ch:Email>contact@example.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:DeviceId>N9K-C9508@C@TXX12345678</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>dc3-test</ch:Name>
<ch:Contact>Jay Tester</ch:Contact> <ch:ContactEmail>contact@example.com</ch:ContactEmail>
<ch:ContactPhoneNumber>+91-80-1234-5678</ch:ContactPhoneNumber>
<ch:StreetAddress>#1, Any Street</ch:StreetAddress> </ch:SystemInfo> </ch:CustomerData>
<ch:Device> <rme:Chassis xmlns:rme="http://www.cisco.com/rme/4.1">
<rme:Model>N9K-C9508</rme:Model>

```

```

<rme:HardwareVersion>0.405</rme:HardwareVersion>
<rme:SerialNumber>TXX12345678</rme:SerialNumber>
</rme:Chassis>
</ch:Device>
</ch:CallHome>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging logfile | tail -n 200</aml-block:Name> <aml-block:Data
encoding="plain">
<![CDATA[2013 May 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages)
cleared by user
2013 May 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16: Invalid
argument: - sshd[14484]
2013 May 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 12:11:18 dc3-test %SYSMGR-STANDBY-5-SUBPROC_TERMINATED: \"System Manager (gsync
controller)\" (PID 12000) has finished with error code SYSMGR_EXITCODE_GSYNCFAILED_NONFATAL
(12).
2013 May 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 16:28:44 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 3504)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:29:08 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23210)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:29:17 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23294)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is becoming
active (pre-start phase).
2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is becoming
active.
2013 May 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinform: mts_send failed -
device_test
2013 May 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR: netstack [4336] Unrecognized
message from MRIB. Major type 1807
2013 May 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN
2013 May 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19072]
2013 May 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19073]

```

Sample Syslog Alert Notification in XML Format

```

2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19079]
2013 May 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP
2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19105]
2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 2 present but all
AC inputs are not connected, ac-redundancy might be affected
2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 3 present but all
AC inputs are not connected, ac-redundancy might be affected
2013 May 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
2013 May 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:30:47 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 4820)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:31:02 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24239)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:31:14 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24401)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
2013 May 17 16:31:23 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24407)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR: netstack [4336] (null)
2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
with component MTS_SAP_ELTm opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1)
]]> </aml-block:Data> </aml-block:Attachment> <aml-block:Attachment type="inline">
<aml-block:Name> <aml-block:Data encoding="plain"> <![CDATA[
dc3-test interfaces:
Ethernet3/1      Ethernet3/2      Ethernet3/3
Ethernet3/4      Ethernet3/5      Ethernet3/6
Ethernet3/7      Ethernet3/8      Ethernet3/9
Ethernet3/10     Ethernet3/11     Ethernet3/12
Ethernet3/13     Ethernet3/14     Ethernet3/15
Ethernet3/16     Ethernet3/17     Ethernet3/18
Ethernet3/19     Ethernet3/20     Ethernet3/21
Ethernet3/22     Ethernet3/23     Ethernet3/24
Ethernet3/25     Ethernet3/26     Ethernet3/27
Ethernet3/28     Ethernet3/29     Ethernet3/30
Ethernet3/31     Ethernet3/32     Ethernet3/33
Ethernet3/34     Ethernet3/35     Ethernet3/36
Ethernet3/37     Ethernet3/38     Ethernet3/39

```



```

Ethernet3/40      Ethernet3/41      Ethernet3/42
Ethernet3/43      Ethernet3/44      Ethernet3/45
Ethernet3/46      Ethernet3/47      Ethernet3/48

]]>
</aml-block:Data>
</aml-block:Attachment>
<aml-block:Attachment type="inline">
<aml-block:Name> <aml-block:Data encoding="plain"> <!--> </aml-block:Data>
</aml-block:Attachment> <aml-block:Attachment type="inline"> <aml-block:Name>show license
usage</aml-block:Name> <aml-block:Data encoding="plain">
<![CDATA[Feature Ins Lic Status Expiry Date Comments
Count
-----
LAN_ENTERPRISE_SERVICES_PKG Yes - Unused Never -
-----
]]>
</aml-block:Data>
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>

```

MIBs

MIBs	MIBs Link
MIBs related to Smart Call Home	To locate and download supported MIBs, go to the following link: https://cisco.github.io/cisco-mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html



CHAPTER 13

Configuring Session Manager

This chapter describes how to configure Session Manager on Cisco NX-OS devices.

This chapter contains the following sections:

- [About Session Manager, on page 213](#)
- [Prerequisites for Session Manager, on page 214](#)
- [Guidelines and Limitations for Session Manager, on page 214](#)
- [Configuring Session Manager, on page 214](#)
- [Verifying the Session Manager Configuration, on page 217](#)
- [Configuration Example for Session Manager, on page 217](#)
- [Additional References, on page 218](#)

About Session Manager

Session Manager allows you to implement your configuration changes in batch mode. Session Manager works in the following phases:

- **Configuration session**—Creates a list of commands that you want to implement in Session Manager mode.
- **Validation**—Provides a basic semantic check on your configuration. Cisco NX-OS returns an error if the semantic check fails on any part of the configuration.
- **Verification**—Verifies the configuration as a whole, based on the existing hardware and software configuration and resources. Cisco NX-OS returns an error if the configuration does not pass this verification phase.
- **Commit**—Cisco NX-OS verifies the complete configuration and applies the changes to the device. If a failure occurs, Cisco NX-OS reverts to the original configuration.
- **Abort**—Discards the configuration changes before implementation.

You can optionally end a configuration session without committing the changes. You can also save a configuration session.

High Availability

Session Manager sessions remain available after a supervisor switchover. Sessions are not persistent across a software reload.

Prerequisites for Session Manager

Make sure that you have the privilege level required to support the Session Manager commands that you plan to use.

Guidelines and Limitations for Session Manager

Session Manager has the following configuration guidelines and limitations:

- Configuration for only one service access point (SAP) can be performed using one session.
- Configuration sessions are not persistent across reloads.
- Session Manager supports only access control list (ACL) and quality of service (QoS) features.
- You can create up to 32 configuration sessions.
- You can configure a maximum of 20,000 commands across all sessions.
- You cannot simultaneously execute configuration commands in more than one configuration session or configuration terminal mode. Parallel configurations (for example, one configuration session and one configuration terminal) could cause validation or verification failures in the configuration session.
- If an interface reloads while you are configuring it in a configuration session, Session Manager can accept the commands even if the interface is not present in the device.

Configuring Session Manager



Note Be aware that the Cisco NX-OS commands might differ from Cisco IOS commands.

Creating a Session

You can create up to 32 configuration sessions.

Procedure

	Command or Action	Purpose
Step 1	configure session <i>name</i> Example: <pre>switch# configure session myACLs switch(config-s)#</pre>	Creates a configuration session and enters session configuration mode. The name can be any alphanumeric string. Displays the contents of the session.
Step 2	(Optional) show configuration session [<i>name</i>] Example: <pre>switch(config-s)# show configuration session myACLs</pre>	Displays the contents of the session.
Step 3	(Optional) save <i>location</i> Example: <pre>switch(config-s)# save bootflash:sessions/myACLs</pre>	Saves the session to a file. The location can be in bootflash:, slot0:, or volatile:.

Configuring ACLs in a Session

You can configure ACLs within a configuration session.

Procedure

	Command or Action	Purpose
Step 1	configure session <i>name</i> Example: <pre>switch# configure session myacls switch(config-s)#</pre>	Creates a configuration session and enters session configuration mode. The name can be any alphanumeric string.
Step 2	ip access-list <i>name</i> Example: <pre>switch(config-s)# ip access-list acl1 switch(config-s-acl)#</pre>	Creates an ACL and enters a configuration mode for that ACL.
Step 3	(Optional) permit <i>protocol source destination</i> Example: <pre>switch(config-s-acl)# permit tcp any any</pre>	Adds a permit statement to the ACL.
Step 4	interface <i>interface-type number</i> Example: <pre>switch(config-s-acl)# interface ethernet 2/1 switch(config-s-if)#</pre>	Enters interface configuration mode.

	Command or Action	Purpose
Step 5	ip access-group <i>name</i> { in out } Example: <pre>switch(config-s-if)# ip access-group acl1 in</pre>	Specifies the direction of traffic the access group is applied to.
Step 6	(Optional) show configuration session [<i>name</i>] Example: <pre>switch(config-s-if)# show configuration session myacls</pre>	Displays the contents of the session.

Verifying a Session

Use the following command in session mode to verify a session:

Command	Purpose
verify [verbose] Example: <pre>switch(config-s)# verify</pre>	Verifies the configuration as a whole, based on the existing hardware and software configuration and resources. Cisco NX-OS returns an error if the configuration does not pass this verification.

Committing a Session

Use the following command in session mode to commit a session:

Command	Purpose
commit [verbose] Example: <pre>switch(config-s)# commit</pre>	Validates the configuration changes made in the current session and applies valid changes to the device. If the validation fails, Cisco NX-OS reverts to the original configuration.

Saving a Session

Use the following command in session mode to save a session:

Command	Purpose
save <i>location</i> Example: <pre>switch(config-s)# save bootflash:sessions/myACLs</pre>	(Optional) Saves the session to a file. The location can be in bootflash:, slot0:, or volatile:.

Discarding a Session

Use the following command in session mode to discard a session:

Command	Purpose
abort Example: switch(config-s)# abort switch#	Discards the configuration session without applying the changes.

Verifying the Session Manager Configuration

To display the Session Manager configuration information, perform one of the following tasks:

Command	Purpose
show configuration session [name]	Displays the contents of the configuration session.
show configuration session status [name]	Displays the status of the configuration session.
show configuration session summary	Displays a summary of all the configuration sessions.

Configuration Example for Session Manager

This example shows how to create and commit an ACL configuration using Session Manager:

```
switch# configure session ACL_tcp_in
Config Session started, Session ID is 1
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-s)# ip access-list ACL1
switch(config-s-acl)# permit tcp any any
switch(config)# interface e 7/1
switch(config-if)# ip access-group ACL1 in
switch(config-if)# exit
switch(config)# exit
switch# config session ACL_tcp_in
Config Session started, Session ID is 1
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-s)# verify
Verification Successful
switch(config-s)# commit
Commit Successful
switch#
```

Additional References

Related Documents

Related Topic	Document Title
Configuration files	<i>Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide</i>



CHAPTER 14

Configuring the Scheduler

This chapter describes how to configure the scheduler on Cisco NX-OS devices.

This chapter includes the following sections:

- [About the Scheduler, on page 219](#)
- [Prerequisites for the Scheduler, on page 220](#)
- [Guidelines and Limitations for the Scheduler, on page 220](#)
- [Default Settings for the Scheduler, on page 221](#)
- [Configuring the Scheduler, on page 221](#)
- [Verifying the Scheduler Configuration, on page 227](#)
- [Configuration Examples for the Scheduler, on page 228](#)

About the Scheduler

The scheduler allows you to define and set a timetable for maintenance activities such as the following:

- Quality of service (QoS) policy changes
- Data backup
- Saving a configuration

Jobs consist of a single command or multiple commands that define routine activities. Jobs can be scheduled one time or at periodic intervals.

The scheduler defines a job and its timetable as follows:

- **Job**—A routine task or tasks defined as a command list and completed according to a specified schedule.
- **Schedule**—The timetable for completing a job. You can assign multiple jobs to a schedule. A schedule is defined as either periodic or one-time only:
 - **Periodic mode**—A recurring interval that continues until you delete the job. You can configure the following types of intervals:
 - **Daily**—A job is completed once a day.
 - **Weekly**—A job is completed once a week.
 - **Monthly**—A job is completed once a month.
 - **Delta**—A job begins at the specified start time and then at specified intervals (days:hours:minutes).

- One-time mode—A job is completed only once at a specified time.

Remote User Authentication

Before starting a job, the scheduler authenticates the user who created the job. Since user credentials from a remote authentication are not retained long enough to support a scheduled job, you need to locally configure the authentication passwords for users who create jobs. These passwords are part of the scheduler configuration and are not considered a locally configured user.

Before starting the job, the scheduler validates the local password against the password from the remote authentication server.

Logs

The scheduler maintains a log file containing the job output. If the size of the job output is greater than the size of the log file, the output is truncated.

High Availability

Scheduled jobs remain available after a supervisor switchover or a software reload.

Prerequisites for the Scheduler

The scheduler has the following prerequisites:

- You must enable any conditional features before you can configure those features in a job.
- You must have a valid license installed for any licensed features that you want to configure in the job.
- You must have network-admin user privileges to configure a scheduled job.

Guidelines and Limitations for the Scheduler

The scheduler has the following configuration guidelines and limitations:

- The scheduler can fail if it encounters one of the following while performing a job:
 - Verify that you have configured the time. The scheduler does not apply a default timetable. If you create a schedule and assign jobs and do not configure the time, the job is not started.
 - While defining a job, verify that no interactive or disruptive commands (for example, **copy bootflash: file ftp: URI**, **write erase**, **reload**, and other similar commands) are specified because the job is started and conducted noninteractively. When a reload job is scheduled for a given time and executed, the switch goes into a boot loop. Hence it should not be used in scheduler configuration.
- The scheduler accepts `start_time` in the past for any schedule with a repeat option in the **time** command under the schedule mode configuration. It then throws a warning that the entered start time is in the past. The `start_time` of any schedule will always remain the same as it was in the beginning, across reboot, and even after reapplying the previous saved configuration.

- Beginning in Cisco NX-OS Release 9.3(5), a second space is included in the output of the scheduler job configuration CLIs.

Previously, the output had only one space before the job configuration CLI:

```
scheduler job name show_fds.  
show clock >> bootflash:show_fds  
^ (single space)
```

Now it has two spaces before the job configuration CLI:

```
scheduler job name show_fds.  
show clock >> bootflash:show_fds  
^^ (two spaces)
```

There is no impact on the functionality of the scheduler in the NX-OS software for configuration replace, ISSU, reload, and so on. But if you are using a script to read the output of the show run command for reading the scheduler component configuration, then you must update the logic in the script to allow for the extra space.

Default Settings for the Scheduler

This table lists the scheduler default settings.

Parameters	Default
Scheduler state	Disabled
Log file size	16 KB

Configuring the Scheduler

Enabling or Disabling the Scheduler

You can enable the scheduler feature so that you can configure and schedule jobs, or you can disable the scheduler feature after it has been enabled.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] feature scheduler Example: switch(config)# feature scheduler	Enables or disables the scheduler.

	Command or Action	Purpose
Step 3	(Optional) show scheduler config Example: <pre>switch(config)# show scheduler config config terminal feature scheduler scheduler logfile size 16 end</pre>	Displays the scheduler configuration.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Defining the Scheduler Log File Size

You can configure the log file size for capturing jobs, schedules, and job output.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	scheduler logfile size <i>value</i> Example: <pre>switch(config)# scheduler logfile size 1024</pre>	Defines the scheduler log file size in kilobytes. The range is from 16 to 1024. The default is 16. Note If the size of the job output is greater than the size of the log file, then the output is truncated.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring Remote User Authentication

You can configure the scheduler to use remote authentication for users who want to configure and schedule jobs.



Note Remote users must authenticate with their clear text password before creating and configuring jobs.



Note Remote user passwords are always shown in encrypted form in the output of the **show running-config** command. The encrypted option (7) in the command supports the ASCII device configuration.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	scheduler aaa-authentication password [0 7] password Example: <pre>switch(config)# scheduler aaa-authentication password X12y34Z56a</pre>	Configures a cleartext password for the user who is currently logged in.
Step 3	scheduler aaa-authentication username name password [0 7] password Example: <pre>switch(config)# scheduler aaa-authentication username newuser password Z98y76X54b</pre>	Configures a cleartext password for a remote user.
Step 4	(Optional) show running-config include "scheduler aaa-authentication" Example: <pre>switch(config)# show running-config include "scheduler aaa-authentication"</pre>	Displays the scheduler password information.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Defining a Job

You can define a job including the job name and the command sequence.



Caution After you define a job, you cannot modify or remove commands. To change the job, you must delete it and create a new one.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	scheduler job name string Example: <pre>switch(config)# scheduler job name backup-cfg switch(config-job)#</pre>	Creates a job and enters the job configuration mode. This example creates a scheduler job named "backup-cfg".
Step 3	<code>command1 ;[command2 ;command3 ;...]</code> Example: <pre>switch(config-job)# copy running-config tftp://1.2.3.4/\${SWITCHNAME}-cfg.\${TIMESTAMP} vrf management switch(config-job)#</pre>	Defines the sequence of commands for the specified job. Separate commands with spaces and semicolons (for example, “;”). This example creates a scheduler job that saves the running configuration to a file in the bootflash. The job then copies the file from the bootflash to a TFTP server and creates the filename using the current timestamp and switch name.
Step 4	(Optional) show scheduler job [name name] Example: <pre>switch(config-job)# show scheduler job</pre>	Displays the job information.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Deleting a Job

You can delete a job from the scheduler.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	no scheduler job name string Example: <pre>switch(config)# no scheduler job name configsave switch(config-job)</pre>	Deletes the specified job and all commands defined within it.
Step 3	(Optional) show scheduler job [name name] Example: <pre>switch(config-job)# show scheduler job name configsave</pre>	Displays the job information.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Defining a Timetable

You can define a timetable in the scheduler to be used with one or more jobs.

If you do not specify the time for the **time** commands, the scheduler assumes the current time. For example, if the current time is March 24, 2013, 22:00 hours, jobs are started as follows:

- For the **time start 23:00 repeat 4:00:00** command, the scheduler assumes a start time of March 24, 2013, 23:00 hours.
- For the **time daily 55** command, the scheduler assumes a start time every day at 22:55 hours.
- For the **time weekly 23:00** command, the scheduler assumes a start time every Friday at 23:00 hours.
- For the **time monthly 23:00** command, the scheduler assumes a start time on the 24th of every month at 23:00 hours.



Note

The scheduler will not begin the next occurrence of a job before the last one completes. For example, you have scheduled a job to be completed at one-minute intervals beginning at 22:00; but the job requires two minutes to complete. The scheduler starts the first job at 22:00, completes it at 22:02, and then observes a one-minute interval before starting the next job at 22:03.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	scheduler schedule name <i>string</i> Example: <pre>switch(config)# scheduler schedule name weekendbackupqos switch(config-schedule)#</pre>	Creates a new schedule and places you in schedule configuration mode for that schedule.
Step 3	job name <i>string</i> Example: <pre>switch(config-schedule)# job name offpeakZoning</pre>	Associates a job with this schedule. You can add multiple jobs to a schedule.
Step 4	time daily <i>time</i> Example: <pre>switch(config-schedule)# time daily 23:00</pre>	Indicates the job starts every day at a designated time specified as HH:MM.
Step 5	time weekly <i>[[dow:]HH:]MM</i> Example: <pre>switch(config-schedule)# time weekly Sun:23:00</pre>	Indicates that the job starts on a specified day of the week. Day of the week (dow) specified as one of the following: • An integer such as 1 = Sunday, 2 = Monday, and so on. • An abbreviation such as Sun = Sunday. The maximum length for the entire argument is 10.
Step 6	time monthly <i>[[dm:]HH:]MM</i> Example: <pre>switch(config-schedule)# time monthly 28:23:00</pre>	Indicates the job starts on a specified day each month (dm). If you specify either 29, 30, or 31, the job is started on the last day of each month.
Step 7	time start { now repeat <i>repeat-interval</i> <i>delta-time</i> [repeat <i>repeat-interval</i>]} Example: <pre>switch(config-schedule)# time start now repeat 48:00</pre>	Indicates the job starts periodically. The start-time format is <code>[[[yyy:]mmm:]dd:]HH]:MM</code>. • delta-time—Specifies the amount of time to wait after the schedule is configured before starting a job. • now—Specifies that the job starts now. • repeat repeat-interval—Specifies the frequency at which the job is repeated. In this example, the job starts immediately and repeats every 48 hours.

	Command or Action	Purpose
Step 8	(Optional) show scheduler config Example: <code>switch(config)# show scheduler config</code>	Displays the scheduler configuration.
Step 9	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Clearing the Scheduler Log File

You can clear the scheduler log file.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	clear scheduler logfile Example: <code>switch(config)# clear scheduler logfile</code>	Clears the scheduler log file.

Verifying the Scheduler Configuration

To display the scheduler configuration information, perform one of the following tasks:

Command	Purpose
show scheduler config	Displays the scheduler configuration.
show scheduler job [name string]	Displays the jobs configured.
show scheduler logfile	Displays the contents of the scheduler log file.
show scheduler schedule [name string]	Displays the schedules configured.

Configuration Examples for the Scheduler

Creating a Scheduler Job

This example shows how to create a scheduler job that saves the running configuration to a file in the bootflash. The job then copies the file from the bootflash to a TFTP server (creates the filename using the current timestamp and switch name):

```
switch# configure terminal
switch(config)# scheduler job name backup-cfg
switch(config-job)# copy running-config
tftp://1.2.3.4/(SWITCHNAME)-cfg.$(TIMESTAMP) vrf management
switch(config-job)# end
switch(config)#
```

Scheduling a Scheduler Job

This example shows how to schedule a scheduler job called backup-cfg to run daily at 1 a.m.:

```
switch# configure terminal
switch(config)# scheduler schedule name daily
switch(config-if)# job name backup-cfg
switch(config-if)# time daily 1:00
switch(config-if)# end
switch(config)#
```

Displaying the Job Schedule

This example shows how to display the job schedule:

```
switch# show scheduler schedule
Schedule Name : daily
-----
User Name : admin
Schedule Type : Run every day at 1 Hrs 00 Mins
Last Execution Time : Fri Jan 2 1:00:00 2013
Last Completion Time: Fri Jan 2 1:00:01 2013
Execution count : 2
-----
Job Name Last Execution Status
-----
back-cfg Success (0)
switch#
```

Displaying the Results of Running Scheduler Jobs

This example shows how to display the results of scheduler jobs that have been executed by the scheduler:

```
switch# show scheduler logfile
Job Name : back-cfg Job Status: Failed (1)
Schedule Name : daily User Name : admin
Completion time: Fri Jan 1 1:00:01 2013
```

```
----- Job Output -----
`cli var name timestamp 2013-01-01-01.00.00`
`copy running-config bootflash:/${(HOSTNAME)}-cfg.${(timestamp)}`
`copy bootflash:/switch-cfg.2013-01-01-01.00.00 tftp://1.2.3.4/ vrf management `
copy: cannot access file '/bootflash/switch-cfg.2013-01-01-01.00.00'
=====
Job Name : back-cfg Job Status: Success (0)
Schedule Name : daily User Name : admin
Completion time: Fri Jan 2 1:00:01 2013
----- Job Output -----
`cli var name timestamp 2013-01-02-01.00.00`
`copy running-config bootflash:/switch-cfg.2013-01-02-01.00.00`
`copy bootflash:/switch-cfg.2013-01-02-01.00.00 tftp://1.2.3.4/ vrf management `
Connection to Server Established.
[ ] 0.50KBTrying to connect to tftp server.....
[##### ] 24.50KB
TFTP put operation was successful
=====
switch#
```




CHAPTER 15

Simple Network Management Protocol

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.



Note The license notifications are enabled by default.

- [SNMP Fundamentals, on page 231](#)
- [Guidelines and Limitations, on page 243](#)
- [Default Settings for SNMP, on page 245](#)
- [Configure SNMP, on page 246](#)
- [Configure SNMP Local Engine ID, on page 269](#)
- [Verify SNMP Configuration, on page 270](#)
- [Configuration Examples, on page 271](#)
- [Additional References, on page 273](#)

SNMP Fundamentals

Framework

The SNMP framework consists of three parts:

- An SNMP manager—The system used to control and monitor the activities of network devices using SNMP.
- An SNMP agent—The software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems. The Cisco Nexus device supports the agent and MIB. To enable the SNMP agent, you must define the relationship between the manager and the agent.
- A managed information base (MIB)—The collection of managed objects on the SNMP agent



Note SNMP is defined in RFCs 3411 to 3418. The device supports SNMPv1, SNMPv2c, and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security.

Cisco NX-OS supports SNMP over IPv6.

Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

Cisco NX-OS generates SNMP notifications as either traps or informs. A trap is an asynchronous, unacknowledged message sent from the agent to the SNMP managers listed in the host receiver table. Informs are asynchronous messages sent from the SNMP agent to the SNMP manager which the manager must acknowledge receipt of.

Traps are less reliable than informs because the SNMP manager does not send any acknowledgment when it receives a trap. The device cannot determine if the trap was received. An SNMP manager that receives an inform request acknowledges the message with an SNMP response protocol data unit (PDU). If the device never receives a response, it can send the inform request again.

You can configure Cisco NX-OS to send notifications to multiple host receivers.

The following table lists the SNMP traps that are enabled by default.

Trap Type	Description
generic	: coldStart
entity	: entity_fan_status_change
entity	: entity_mib_change
entity	: entity_module_status_change
entity	: entity_module_inserted
entity	: entity_module_removed
entity	: entity_power_out_change
entity	: entity_power_status_change
entity	: entity_unrecognised_module
link	: cErrDisableInterfaceEventRev1
link	: cieLinkDown
link	: cieLinkUp
link	: cmn-mac-move-notification
link	: delayed-link-state-change
link	: extended-linkDown
link	: extended-linkUp

Trap Type	Description
link	: linkDown
link	: linkUp
rf	: redundancy_framework
upgrade	: UpgradeOpNotifyOnCompletion
upgrade	: UpgradeJobStatusNotify
entity	: entity_sensor
rmon	: fallingAlarm
rmon	: hcRisingAlarm
rmon	: hcFallingAlarm
rmon	: risingAlarm

Security Levels and Models

A security model is an authentication strategy that is set up for a user and the role in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet. SNMPv3 provides for both security models and security levels.

The security level determines if an SNMP message needs to be protected from disclosure and if the message needs to be authenticated. The various security levels that exist within a security model are as follows:

- noAuthNoPriv—Security level that does not provide authentication or encryption. This level is not supported for SNMPv3.
- authNoPriv—Security level that provides authentication but does not provide encryption.
- authPriv—Security level that provides both authentication and encryption.

The security models available are:

- SNMPv1
- SNMPv2c
- SNMPv3

The following table identifies what the combinations of security models and levels mean.

Table 14: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.

Model	Level	Authentication	Encryption	What Happens
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	authNoPriv	HMAC-MD5, HMAC-SHA, or SHA-256	No	Provides authentication based on the Hash-Based Message Authentication Code (HMAC) Message Digest 5 (MD5) algorithm or the HMAC Secure Hash Algorithm (SHA).
v3	authPriv	HMAC-MD5, HMAC-SHA, or SHA-256	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides Data Encryption Standard (DES) 56-bit encryption in addition to authentication based on the Cipher Block Chaining (CBC) DES (DES-56) standard.

SNMP Logical Entities

A device can support multiple instances of a logical network entity, such as protocol instances or virtual routing and forwarding (VRF) instances. Most existing MIBs cannot distinguish between the multiple logical network entities. For example, the original OSPF-MIB assumes a single protocol instance on a device, but you can now configure multiple OSPF instances on a device.

Context

An SNMP context is a collection of management information that you can access through the SNMP agent.

Groups

SNMP access rights are organized by groups. Each group in SNMP is similar to a role through the CLI. Each group is defined with read access or read-write access.

You can begin communicating with the agent once your username is created, your roles are set up by your administrator, and you are added to the roles.



Note Because *group* is a standard SNMP term used industry-wide, we refer to roles as groups in this SNMP section.

Embedded Event Manager (EEM)

The Embedded Event Manager (EEM) feature monitors events, including SNMP MIB objects, and triggers an action based on these events. One of the actions could be to send an SNMP notification. EEM sends the cEventMgrPolicyEvent of CISCO-EMBEDDED-EVENT-MGR-MIB as the SNMP notification.

VRF Support

SNMP is also VRF aware. You can configure SNMP to use a particular VRF to reach the SNMP notification host receiver. You can also configure SNMP to filter notifications to an SNMP host receiver based on the VRF where the notification occurred.

Cisco NX-OS Support

Cisco NX-OS supports one instance of the SNMP. Cisco NX-OS supports stateless restarts for SNMP. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration. Cisco NX-OS supports SNMP over IPv6.

Multiple Instance Support

SNMP supports multiple MIB module instances and maps them to logical network entities. Cisco NX-OS supports the CISCO-CONTEXT-MAPPING-MIB to map between SNMP contexts and logical network entities. You can associate an SNMP context to a VRF, protocol instance, or topology.

SNMPv3 - uses contexts to distinguish between these multiple instances. A device can support multiple contexts for different logical network entities. An SNMP context allows the SNMP manager to access one of the multiple instances of a MIB module supported on the device for the different logical network entities. SNMPv3 supports contexts with the contextName field of the SNMPv3 PDU. You can map this contextName field to a particular protocol instance or VRF.

SNMPv2c - you can map the SNMP community to a context using the snmpCommunityContextName MIB object in the SNMP-COMMUNITY-MIB (RFC 3584). You can then map this snmpCommunityContextName to a particular protocol instance or VRF using the CISCO-CONTEXT-MAPPING-MIB or the CLI.



Note SNMP is defined in RFCs 3411 to 3418. The device supports SNMPv1, SNMPv2c, and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security.

SNMP Functional Overview

The SNMP framework consists of three parts:

- An SNMP manager—The system used to control and monitor the activities of network devices using SNMP.
- An SNMP agent—The software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems. The Cisco Nexus device supports the agent and MIB. To enable the SNMP agent, you must define the relationship between the manager and the agent.
- A managed information base (MIB)—The collection of managed objects on the SNMP agent

SNMP is defined in RFCs 3411 to 3418.

The device supports SNMPv1, SNMPv2c, and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security.

Cisco NX-OS supports SNMP over IPv6.

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

Cisco NX-OS generates SNMP notifications as either traps or informs. A trap is an asynchronous, unacknowledged message sent from the agent to the SNMP managers listed in the host receiver table. Informs are asynchronous messages sent from the SNMP agent to the SNMP manager which the manager must acknowledge receipt of.

Traps are less reliable than informs because the SNMP manager does not send any acknowledgment when it receives a trap. The device cannot determine if the trap was received. An SNMP manager that receives an inform request acknowledges the message with an SNMP response protocol data unit (PDU). If the device never receives a response, it can send the inform request again.

You can configure Cisco NX-OS to send notifications to multiple host receivers.

The following table lists the SNMP traps that are enabled by default.

Trap Type	Description
generic	: coldStart
entity	: entity_fan_status_change
entity	: entity_mib_change
entity	: entity_module_status_change
entity	: entity_module_inserted
entity	: entity_module_removed
entity	: entity_power_out_change
entity	: entity_power_status_change
entity	: entity_unrecognised_module
link	: cErrDisableInterfaceEventRev1
link	: cieLinkDown
link	: cieLinkUp
link	: cmn-mac-move-notification
link	: delayed-link-state-change
link	: extended-linkDown
link	: extended-linkUp
link	: linkDown
link	: linkUp
rf	: redundancy_framework

Trap Type	Description
upgrade	: UpgradeOpNotifyOnCompletion
upgrade	: UpgradeJobStatusNotify
entity	: entity_sensor
rmon	: fallingAlarm
rmon	: hcRisingAlarm
rmon	: hcFallingAlarm
rmon	: risingAlarm

SNMPv3

SNMPv3 provides secure access to devices by a combination of authenticating and encrypting frames over the network. The SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages have not been altered or destroyed in an unauthorized manner and that data sequences have not been altered to an extent greater than can occur nonmaliciously.
- Authentication—Ensures that the claimed identity of the user on whose behalf received data was originated is confirmed.
- Encryption—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMPv3 authorizes management operations only by configured users and encrypts SNMP messages.

Authentication Protocols

Cisco NX-OS uses three authentication protocols for SNMPv3:

- HMAC-MD5-96 authentication protocol
- HMAC-SHA-96 authentication protocol
- SHA-256 authentication protocol



Note Beginning with Cisco NX-OS release 9.3(7), HMAC-SHA-256 authentication protocol is used for SNMPv3. When SHA-256 SNMP users are configured on the switch, ISSD is recommended by **install all** cmd else there will be config loss.

Cisco NX-OS uses Advanced Encryption Standard (AES) as one of the privacy protocols for SNMPv3 message encryption and conforms with RFC 3826.

The **priv** option offers a choice of DES or 128-bit AES encryption for SNMP security encryption. The **priv** option and the **aes-128** token indicate that this privacy password is for generating a 128-bit AES key. The AES priv password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 case-sensitive, alphanumeric characters. If you use the localized key, you can specify a maximum of 130 characters.



Note For an SNMPv3 operation using the external AAA server, you must use AES for the privacy protocol in the user configuration on the external AAA server.

User Configuration Synchronization

SNMPv3 user management can be centralized at the Access Authentication and Accounting (AAA) server level. This centralized user management allows the SNMP agent in Cisco NX-OS to leverage the user authentication service of the AAA server. Once user authentication is verified, the SNMP PDUs are processed further. Additionally, the AAA server is also used to store user group names. SNMP uses the group names to apply the access/role policy that is locally available in the switch.

Any configuration changes made to the user group, role, or password results in database synchronization for both SNMP and AAA.

Cisco NX-OS synchronizes the user configuration in the following ways:

- The authentication passphrase specified in the **snmp-server user** command becomes the password for the CLI user.
- The password specified in the **username** command becomes the authentication and privacy passphrases for the SNMP user.
- If you create or delete a user using either SNMP or the CLI, the user is created or deleted for both SNMP and the CLI.
- User-role mapping changes are synchronized in SNMP and the CLI.
- Role changes (deletions or modifications) from the CLI are synchronized to SNMP.



Note When you configure a passphrase/password in localized key/encrypted format, Cisco NX-OS does not synchronize the user information (passwords, roles, and so on). Cisco NX-OS holds the synchronized user configuration for 60 minutes by default.

Security Models and Levels for SNMPv1, v2, v3

The security level determines if an SNMP message needs to be protected from disclosure and if the message needs to be authenticated. The various security levels that exist within a security model are as follows:

- **noAuthNoPriv**—Security level that does not provide authentication or encryption. This level is not supported for SNMPv3.
- **authNoPriv**—Security level that provides authentication but does not provide encryption.
- **authPriv**—Security level that provides both authentication and encryption.

Three security models are available: SNMPv1, SNMPv2c, and SNMPv3. The security model combined with the security level determine the security mechanism applied when the SNMP message is processed. The following table identifies what the combinations of security models and levels mean.

Table 15: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	authNoPriv	HMAC-MD5, HMAC-SHA, or SHA-256	No	Provides authentication based on the Hash-Based Message Authentication Code (HMAC) Message Digest 5 (MD5) algorithm or the HMAC Secure Hash Algorithm (SHA).
v3	authPriv	HMAC-MD5, HMAC-SHA, or SHA-256	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides Data Encryption Standard (DES) 56-bit encryption in addition to authentication based on the Cipher Block Chaining (CBC) DES (DES-56) standard.

User-Based Security Model

The SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages have not been altered or destroyed in an unauthorized manner and that data sequences have not been altered to an extent greater than can occur nonmaliciously.
- Message origin authentication—Ensures that the claimed identity of the user on whose behalf received data was originated is confirmed.
- Message confidentiality—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMPv3 authorizes management operations only by configured users and encrypts SNMP messages.

Cisco NX-OS uses three authentication protocols for SNMPv3:

- HMAC-MD5-96 authentication protocol
- HMAC-SHA-96 authentication protocol
- SHA-256 authentication protocol

Beginning with Cisco NX-OS release 9.3(7), HMAC-SHA-256 authentication protocol is used for SNMPv3.



Note When SHA-256 SNMP users are configured on the switch, ISSD is recommended by **install all** cmd else there will be config loss.

Cisco NX-OS uses Advanced Encryption Standard (AES) as one of the privacy protocols for SNMPv3 message encryption and conforms with RFC 3826.

The **priv** option offers a choice of DES or 128-bit AES encryption for SNMP security encryption. The **priv** option and the **aes-128** token indicate that this privacy password is for generating a 128-bit AES key. The AES priv password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 case-sensitive, alphanumeric characters. If you use the localized key, you can specify a maximum of 130 characters.



Note For an SNMPv3 operation using the external AAA server, you must use AES for the privacy protocol in the user configuration on the external AAA server.

CLI and SNMP User Synchronization

SNMPv3 user management can be centralized at the Access Authentication and Accounting (AAA) server level. This centralized user management allows the SNMP agent in Cisco NX-OS to leverage the user authentication service of the AAA server. Once user authentication is verified, the SNMP PDUs are processed further. Additionally, the AAA server is also used to store user group names. SNMP uses the group names to apply the access/role policy that is locally available in the switch.

Any configuration changes made to the user group, role, or password results in database synchronization for both SNMP and AAA.

Cisco NX-OS synchronizes the user configuration in the following ways:

- The authentication passphrase specified in the **snmp-server user** command becomes the password for the CLI user.
- The password specified in the **username** command becomes the authentication and privacy passphrases for the SNMP user.
- If you create or delete a user using either SNMP or the CLI, the user is created or deleted for both SNMP and the CLI.
- User-role mapping changes are synchronized in SNMP and the CLI.
- Role changes (deletions or modifications) from the CLI are synchronized to SNMP.



Note When you configure a passphrase/password in localized key/encrypted format, Cisco NX-OS does not synchronize the user information (passwords, roles, and so on).

Cisco NX-OS holds the synchronized user configuration for 60 minutes by default.

Disable Security and SNMP User Synchronization

Beginning with Cisco NX-OS Release 10.2(2)F, the following desynchronization command is introduced to provide you an option to disable the user synchronization between the SNMP and the security (AAA or CLI) components:

snmp-server disable snmp-aaa sync

You can execute this command from the configure terminal on the Nexus switches. By default, the **no** form of the desynchronization command is available on the switch.

When the no-form of the desynchronization command is enabled on the device, for example, `switch (config)# no snmp-server disable snmp-aaa sync`, a user created through **snmp-server user** CLI results in the creation of a **username** CLI for that user in the running configuration and conversely. So, the user can log in to the switch, using the authentication credentials mentioned in the **snmp-server user** CLI or the **username** CLI, at the time of creation/update, and will also be able to perform SNMP operations from a network manager on the switch. Thus, the **no** form of the desynchronization command ensures that the user synchronization between the SNMP and the AAA functions the way it did in the releases prior to 10.2(2)F.

When the desynchronization command is enabled on the device, for example, `switch (config)# snmp-server disable snmp-aaa sync`, a user created through the **snmp-server user** command does not create a username configuration for that user. So, the user cannot log in to the switch and is only allowed to do SNMP operations through a network manager on the switch. Similarly, creation of a security user through the **username** CLI does not create a corresponding **snmp-server user** CLI for the user. This user will be able to log in to the switch but will not be able to perform any SNMP operation on the switch. This is a new feature that the desynchronization command has introduced from Release 10.2(2)F.

You can view the status of the desynchronization command in one of the following ways:

- The value of the field `SNMP-AAA sync disable` in the output of the CLI **show snmp internal globals**
- The value of the field `disableSnmpAaaSync` in the `sys/snmp/inst/globals MO`
- The CLI print in the **show-running-config** output and **show-running-config-snmp** output or **show-running-all** output, based on whether the command is enabled or disabled, respectively

Remote Users

With regard to remote users, who are authenticated for login through external servers using protocols such as RADIUS and TACACS+, when the desynchronization command is enabled on the switch, the remote users cannot be created in SNMP. For more information, refer to the *Configuring AAA* chapter in the *Cisco Nexus 9000 NX-OS Security Configuration Guide*.

However, when the **no** form of the desynchronization command is enabled on the switch, if a remote user is created in AAA, the corresponding user is created in SNMP as well. Furthermore, the user will not be available in the running-config output of SNMP, but will be able to perform SNMP operations on the managed device, which is an existing feature prior to Release 10.2(2)F. However, external NMS client's originated snmpwalk

and so on does not go through unless the **snmp-sync-password** command is executed in the device for the remote user created in AAA.

DCNM Security Users

The security users created using DCNM (also called as Nexus Dashboard Fabric Controller from Release 12.0.1a onwards) will not have a corresponding SNMPv3 profile when the desynchronization command is enabled. When the synchronization is disabled, the users created on the security component can log in to the switch, but the switches will not be discovered by the controller, as the controller uses the SNMP configuration created for the security user to discover the switch. Furthermore, the SNMP does not recognize the security users created due to the desynchronized state of the userDB, resulting in failure to discover the switch. Therefore, to have the switches discovered by the controller, the SNMP user must be explicitly created. It is not recommended to use the desynchronization command along with DCNM functionality. For more information, refer to the *Cisco Nexus 9000 NX-OS Security Configuration Guide*.

ISSD and ISSU

In general, if SNMP user synchronization has been disabled, do not enable SNMP user synchronization unless all the desynchronized users are removed. A running configuration with such a combination will result in a configuration replace failure.

The only way to achieve the desynchronized state in older releases without the desynchronization command is as follows:

- If the Disruptive/ND-ISSD is performed from a desynchronized state to a release without the desynchronization command, the desynchronized databases will be ported as-is through ISSD to the previous release.



Note Any modifications done to the user database after such ISSD will be synchronized between SNMP and security components.

After such ISSD, ISSU to a release with desynchronization command brings in the desynchronized user database as-is, but the desynchronization command comes up in its default **no** form. If required, enable the desynchronization command.

Group-Based SNMP Access



Note Because *group* is a standard SNMP term used industry-wide, we refer to roles as groups in this SNMP section.

SNMP access rights are organized by groups. Each group in SNMP is similar to a role through the CLI. Each group is defined with read access or read-write access.

You can begin communicating with the agent once your username is created, your roles are set up by your administrator, and you are added to the roles.

SNMP and Embedded Event Manager

The Embedded Event Manager (EEM) feature monitors events, including SNMP MIB objects, and triggers an action based on these events. One of the actions could be to send an SNMP notification. EEM sends the `cEventMgrPolicyEvent` of `CISCO-EMBEDDED-EVENT-MGR-MIB` as the SNMP notification.

Multiple Instance Support

A device can support multiple instances of a logical network entity, such as protocol instances or virtual routing and forwarding (VRF) instances. Most existing MIBs cannot distinguish between these multiple logical network entities. For example, the original OSPF-MIB assumes a single protocol instance on a device, but you can now configure multiple OSPF instances on a device.

SNMPv3 uses contexts to distinguish between these multiple instances. An SNMP context is a collection of management information that you can access through the SNMP agent. A device can support multiple contexts for different logical network entities. An SNMP context allows the SNMP manager to access one of the multiple instances of a MIB module supported on the device for the different logical network entities.

Cisco NX-OS supports the `CISCO-CONTEXT-MAPPING-MIB` to map between SNMP contexts and logical network entities. You can associate an SNMP context to a VRF, protocol instance, or topology.

SNMPv3 supports contexts with the `contextName` field of the SNMPv3 PDU. You can map this `contextName` field to a particular protocol instance or VRF.

For SNMPv2c, you can map the SNMP community to a context using the `snmpCommunityContextName` MIB object in the `SNMP-COMMUNITY-MIB` (RFC 3584). You can then map this `snmpCommunityContextName` to a particular protocol instance or VRF using the `CISCO-CONTEXT-MAPPING-MIB` or the CLI.

High Availability for SNMP

Cisco NX-OS supports stateless restarts for SNMP. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

Virtualization Support for SNMP

Cisco NX-OS supports one instance of the SNMP. SNMP supports multiple MIB module instances and maps them to logical network entities.

SNMP is also VRF aware. You can configure SNMP to use a particular VRF to reach the SNMP notification host receiver. You can also configure SNMP to filter notifications to an SNMP host receiver based on the VRF where the notification occurred.

Guidelines and Limitations

SNMP has the following configuration guidelines and limitations:

- Commands configured using SNMP SET should be deleted using SNMP SET only. Commands configured using Command Line Interface (CLI) or NX-API should be deleted using CLI or NX-API only.

- When you create or edit a user in AAA using clear text password, SNMP creates or edits the user to have default auth (md5) and priv types.

When you create or edit a user in SNMP using clear text password, AAA creates or edits the user to have default password type (type 5).

- Access control list (ACLs) can be applied only to local SNMPv3 users configured on the switch. ACLs cannot be applied to remote SNMPv3 users stored on Authentication, Authorization, and Accounting (AAA) servers.
- Do not enable SNMP user synchronisation after it has been disabled unless all desynchronised users are removed. A running configuration with such a combination will result in a configuration replace failure.
- Cisco NX-OS supports read-only access to some SNMP MIBs. See the Cisco NX-OS MIB support list at the following URL for more information: <https://cisco.github.io/cisco-mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html>
- Cisco NX-OS does not support the SNMPv3 noAuthNoPriv security level.
- Cisco Nexus 9000 Series switches and the Cisco Nexus 3164Q, 31128PQ, 3232C, and 3264Q switches support the configuration of the SNMP local engine ID.
- For a nondisruptive downgrade path to an earlier release, if a local engine ID has been configured, then you must unconfigure the local engine ID, and then reconfigure the SNMP users and the community strings.
- Special characters @ and % are not allowed in the SNMP community string.
- The default SNMP PDU value is 1500 bytes. The SNMP agent drops any response PDU that is greater than 1500 bytes, causing the SNMP request to fail. To receive MIB data values larger than 1500 bytes, use the **snmp-server packetsize <byte-count>** command to reconfigure the packet size. The valid byte-count range is from 484 to 17382. When a GETBULK response exceeds the packet size, the data can get truncated.
- You must use either the CLI or SNMP to configure a feature on your switch. Do not configure a feature using both interfaces to the switch.
- Using `cefcFanTrayOperStatus snmpwalk` on an individual fan OID tree where the fan is not populated in chassis, can return a response for next OID entry in the tree. To prevent this behavior, use the `-CI` option in `snmpwalk`.
The behavior is not seen when polling parent OID, or when using `getmany`.
- Cisco Nexus 9000 series switches support upto 10000 flash files for `snmpwalk` request.
- There must be at least one running BGP instance to have full, proper functional behavior of SNMP traps. Configure a BGP routing instance before configuring any `snmp-server traps` related commands.
- Beginning with Release 10.1(1), AES-128 is the recommended encryption algorithm, as it is a strong encryption algorithm. However, DES encryption is also supported.
Downgrade: In-Service System Downgrade (ISSD) with **install all** command is aborted if users with DES privacy protocol are present in the SNMP database. Users need to be reconfigured (using the default AES-128) or deleted. In case of a cold reboot, the SNMP users with DES are deleted.
- When engine ID is configured after configuring the SNMP user, ensure that you perform the following action:

- After changing the engine ID, reconfigure the SNMP user and the related configuration including group, ACL, along with the password. This avoids authentication failure and impact on the ACL and group attached to the user.
- Beginning with Cisco NX-OS Release 10.3(1)F, SNMP (MIBs – 400G Optic MIB, Switch MIB, Datapath MIBs, Interface MIB) is supported on the Cisco Nexus 9808 platform switches.
- The SVI stats are polled only at an interval of every 120 seconds for SNMP cache.
- Beginning with Cisco NX-OS Release 10.3(3)F, Type-6 encryption for SNMPv3 user password is supported with following limitations:
 - Type-6 encryption is successful only if the following is taken care:
 - **feature password encryption aes {tam}** is enabled.
 - Primary key is configured.
 - The **pwd_type 6** option is specified during SNMPv3 user configuration.
- Changing the primary key configuration results in SNMP re-encrypting all Type-6 users stored in its database. However, the SNMP functionalities continue to work the same way as before.
- Primary key configuration is local to the switch. If the user takes the Type-6 configured running data from one switch and applies it on other switch where a different primary key is configured, SNMP features for the same user might not work on the other switch.
- If Type-6 is configured, ensure to remove the configuration, or reconfigure the Type-6 option before downgrading to the release where Type-6 is not supported.
- In case of ISSU, if you migrate from an earlier image (where localizedkey, localizedV2key config is present) to a new image where Type-6 encryption is supported, SNMP won't convert the existing keys to Type-6 encryption.
- Conversion between existing SALT encryption to Type-6 encryption is supported using the **encryption re-encrypt obfuscated** command.
- ASCII-based reloads through disruptive upgrades and **reload-ascii** commands leads to loss of primary key which would impact the SNMP functionality for the Type-6 users.
- If a user enforces re-encryption using the **encryption re-encrypt obfuscated** command, then SNMP encrypts all passwords from non-Type-6 SNMP users to Type-6 mode.



Note The SNMP does not support the **encryption delete type6** command and a syslog warning message is also displayed indicating the same.

Default Settings for SNMP

The following table lists the default settings for SNMP parameters.

Parameters	Default
License notifications	Enabled

Configure SNMP



Note Be aware that the Cisco NX-OS commands for this feature may differ from those commands used in Cisco IOS.



Note From Cisco NX-OS release 9.3(7), HMAC-SHA-256 authentication protocol is used for SNMPv3.

The following section includes the various processes involved to configure SNMP.

Configure SNMP Users

You can configure a user for SNMP.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server user <i>name</i> [pwd_type 6] [auth {md5 sha sha-224 sha-256 sha-384 sha-512} passphrase [auto] [priv [aes-128] passphrase] [engineID <i>id</i>] [localizedkey] [localizedV2key]] Example: <pre>switch(config)# snmp-server user Admin pwd_type 6 auth sha abcd1234 priv abcdefgh</pre>	Configures an SNMP user with authentication and privacy parameters. The passphrase can be any case-sensitive, alphanumeric string up to 64 characters. If you use the localizedkey keyword, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters. sha - Use the HMAC SHA-1 algorithm for authentication. sha-224 - Use the HMAC SHA-224 algorithm for authentication. sha-256 - Use the HMAC SHA-256 algorithm for authentication. sha-384 - Use the HMAC SHA-384 algorithm for authentication.

	Command or Action	Purpose
		sha-512 - Use the HMAC SHA-512 algorithm for authentication. localizedkey - If you use the localizedkey keyword, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters. Instead of plain-text password, hashed password (copied either from the show running config command or generated offline using snmpv3 based open source hash generator tool, see Generate Hashed Password Offline, on page 248) can be configured using the localizedkey keyword. Note When using a localized key, add 0x before the hash value, for example, 0x84a716329158a97ac9f22780629bc26c. localizedV2key - If the localizedV2key is used, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters, without 0x at the beginning. Collect the localizedv2key using show run command, as this is an encrypted data and cannot be generated offline. The engineID format is a 12-digit, colon-separated decimal number. Note - Beginning with Cisco NX-OS Release 10.1(1), AES-128 is the default privacy protocol for SNMPv3. - Beginning with Cisco NX-OS Release 10.3(3)F, the pwd_type 6 keyword is supported to provide Type-6 encryption for SNMP users password.
Step 3	(Optional) show snmp user Example: <pre>switch(config)# show snmp user</pre>	Displays information about one or more SNMP users.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Generate Hashed Password Offline

Perform the following steps to generate hashed password offline, using snmpv3-based open source hash generator tool:



Note The IDs mentioned in this procedure are only sample IDs, the purpose of which is only to explain the procedure better.

1. Get the SNMP engineID from the switch.

```
switch# show snmp engineID
```

Sample output:

```
Local SNMP engineID: [Hex] 8000000903D4C93CEA31CC
[Dec] 128:000:000:009:003:212:201:060:234:049:204
```

2. Use an SNMPv3 based open source hash generator to generate offline hashed password.

```
Linux$ snmpv3-hashgen --auth Hello123 --engine 8000000903D4C93CEA31CC --user1 --mode priv
--hash md5
```

Sample output:

```
User: user1
Auth: Hello123 / 84a716329158a97ac9f22780629bc26c
Priv: Hello123 / 84a716329158a97ac9f22780629bc26c
Engine: 8000000903D4C93CEA31CC
ESXi USM String: u1/84a716329158a97ac9f22780629bc26c/84a716329158a97ac9f22780629bc26c/priv
```

3. Use the auth and priv values to configure the password on the switch.

```
snmp-server user user1 auth md5 0x84a716329158a97ac9f22780629bc26c priv des
0x84a716329158a97ac9f22780629bc26c localizedkey
```

Enforce SNMP Message Encryption

You can configure SNMP to require authentication or encryption for incoming requests. By default, the SNMP agent accepts SNMPv3 messages without authentication and encryption. When you enforce privacy, Cisco NX-OS responds with an authorization error for any SNMPv3 PDU request using a security level parameter of either **noAuthNoPriv** or **authNoPriv**.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server user <i>name</i> enforcePriv Example:	Enforces SNMP message encryption for this user.

	Command or Action	Purpose
	<code>switch(config)# snmp-server user Admin enforcePriv</code>	
Step 3	snmp-server globalEnforcePriv Example: <code>switch(config)# snmp-server globalEnforcePriv</code>	Enforces SNMP message encryption for all users.
Step 4	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Assign SNMPv3 Users to Multiple Roles

After you configure an SNMP user, you can assign multiple roles for the user.



Note Only users belonging to a network-admin role can assign roles to other users.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	snmp-server user <i>name group</i> Example: <code>switch(config)# snmp-server user Admin superuser</code>	Associates this SNMP user with the configured user role.
Step 3	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Create SNMP Communities

You can create SNMP communities for SNMPv1 or SNMPv2c.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server community name {group group ro rw} Example: <pre>switch(config)# snmp-server community public ro</pre>	Creates an SNMP community string.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Filter SNMP Requests

You can assign an access control list (ACL) to an SNMPv2 community to filter SNMP requests. If the assigned ACL allows the incoming request packet, SNMP processes the request. If the ACL denies the request, SNMP drops the request and sends a system message.

Create the ACL with the following parameters:

- Source IP address
- Destination IP address
- Source port
- Destination port
- Protocol (UDP or TCP)

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server community name [use-ipv4acl acl-name] Example:	Assigns an IPv4 ACL to an SNMPv2 community to filter SNMP requests.

	Command or Action	Purpose
	switch(config)# snmp-server community public use-ipv4acl myacl	
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configure SNMP Notification Receivers

You can configure Cisco NX-OS to generate SNMP notifications to multiple host receivers.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	snmp-server host ip-address traps version 1 community [udp_port number] Example: switch(config)# snmp-server host 192.0.2.1 traps version 1 public	Configures a host receiver for SNMPv1 traps. The <i>ip-address</i> can be an IPv4 or IPv6 address. The <i>community</i> can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.
Step 3	snmp-server host ip-address {traps informs} version 2c community [udp_port number] Example: switch(config)# snmp-server host 192.0.2.1 informs version 2c public	Configures a host receiver for SNMPv2c traps or informs. The <i>ip-address</i> can be an IPv4 or IPv6 address. The <i>community</i> can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535.
Step 4	snmp-server host ip-address {traps informs} version 3 {auth noauth priv} username [udp_port number] Example: switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS	Configures a host receiver for SNMPv3 traps or informs. The <i>ip-address</i> can be an IPv4 or IPv6 address. The <i>username</i> can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. Note The SNMP manager must know the user credentials (authKey/PrivKey) based on the SNMP engine ID of the Cisco NX-OS device to authenticate and decrypt the SNMPv3 messages.
Step 5	(Optional) copy running-config startup-config Example:	Copies the running configuration to the startup configuration.

	Command or Action	Purpose
	<code>switch(config)# copy running-config startup-config</code>	

Configure Source Interface for SNMP Notifications

You can configure SNMP to use the IP address of an interface as the source IP address for notifications. When a notification is generated, its source IP address is based on the IP address of this configured interface.

You can configure a source interface as follows:

- All notifications sent to all SNMP notification receivers.
- All notifications sent to a specific SNMP notification receiver. This configuration overrides the global source interface configuration.



Note

Configuring the source interface IP address for outgoing trap packets does not guarantee that the device will use the same interface to send the trap. The source interface IP address defines the source address inside of the SNMP trap, and the connection is opened with the address of the egress interface as source.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server host ip-address source-interface if-type if-number traps version 2c name Example: <pre>snmp-server host 192.0.2.1 source-interface ethernet 2/1 traps version 2c public</pre>	(Optional) Send Traps messages to this host. The traps version is the SNMP version to use for notification messages. 2c indicates that SNMPv2c is to be used.
Step 3	snmp-server host ip-address source-interface if-type if-number use-vrf vrf-name Example: <pre>snmp-server host 192.0.2.1 source-interface ethernet 2/1 use-vrf default</pre>	Configures SNMP to use the selected VRF to communicate with the host receiver. The ip-address can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 32 characters. Note This command does not remove the host configuration.
Step 4	snmp-server host ip-address source-interface if-type if-number [udp_port number]	Configures a host receiver for SNMPv2c traps or informs. The ip-address can be an IPv4 or

	Command or Action	Purpose
	Example: <pre>switch(config)# snmp-server host 192.0.2.1 source-interface ethernet 2/1</pre>	IPv6 address. Use ? to determine the supported interface types. The UDP port number range is from 0 to 65535. This configuration overrides the global source interface configuration.
Step 5	snmp-server source-interface {traps informs} if-type if-number Example: <pre>switch(config)# snmp-server source-interface traps ethernet 2/1</pre>	Configures a source interface for sending out SNMPv2c traps or informs. Use ? to determine the supported interface types.
Step 6	show snmp source-interface Example: <pre>switch(config)# show snmp source-interface</pre>	Displays information about configured source interfaces.

Configure Notification Target User

You must configure a notification target user on the device to send SNMPv3 inform notifications to a notification host receiver.

Cisco NX-OS uses the credentials of the notification target user to encrypt the SNMPv3 inform notification messages to the configured notification host receiver.



Note For authenticating and decrypting the received inform PDU, the notification host receiver should have the same user credentials as configured in Cisco NX-OS to authenticate and decrypt the informs.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server user name [auth {md5 sha sha-256} passphrase [auto] [priv passphrase] [engineID id] Example: <pre>switch(config)# snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID 00:00:00:63:00:01:00:10:20:15:10:03</pre>	Configures the notification target user with the specified engine ID for the notification host receiver. The engine ID format is a 12-digit colon-separated decimal number. Note Beginning with Release 10.1(1), AES-128 is the default privacy protocol for SNMPv3.

	Command or Action	Purpose
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure SNMP Notification Receivers with VRFs

SNMP adds entries into the cExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MIB when you configure the VRF reachability and filtering options for an SNMP notification receiver.



Note You must configure the host before configuring the VRF reachability or filtering options.

You can configure Cisco NX-OS to use a configured VRF to reach the host receiver or to filter notifications based on the VRF in which the notification occurred.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] snmp-server host ip-address use-vrf vrf-name [udp_port number] Example: <pre>switch(config)# snmp-server host 192.0.2.1 use-vrf Blue</pre>	Configures SNMP to use the selected VRF to communicate with the host receiver. The <i>ip-address</i> can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. This command adds an entry into the ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB. The no form of this command removes the VRF reachability information for the configured host and removes the entry from the ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB. Note This command does not remove the host configuration.
Step 3	[no] snmp-server host ip-address filter-vrf vrf-name [udp_port number] Example:	Filters notifications to the notification host receiver based on the configured VRF. The <i>ip-address</i> can be an IPv4 or IPv6 address. The

	Command or Action	Purpose
	<pre>switch(config)# snmp-server host 192.0.2.1 filter-vrf Red</pre>	VRF name can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. This command adds an entry into the ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB. The no form of this command removes the VRF filter information for the configured host and removes the entry from the ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB. Note This command does not remove the host configuration.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure SNMP to Send Traps Using an Inband Port

You can configure SNMP to send traps using an inband port. To do so, you must configure the source interface (at the global or host level) and the VRF used to send the traps.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server source-interface traps <i>if-type if-number</i> Example: <pre>switch(config)# snmp-server source-interface traps ethernet 1/2</pre>	Globally configures a source interface for sending out SNMP traps. Use ? to determine the supported interface types. You can configure the source interface at the global level or at a host level. When the source interface is configured globally, any new host configuration uses the global configuration to send the traps. Note

	Command or Action	Purpose
		To configure a source interface at the host level, use the snmp-server host ip-address source-interface if-type if-number command.
Step 3	(Optional) show snmp source-interface Example: <pre>switch(config)# show snmp source-interface</pre>	Displays information about configured source interfaces.
Step 4	snmp-server host ip-address use-vrf vrf-name [udp_port number] Example: <pre>switch(config)# snmp-server host 171.71.48.164 use-vrf default</pre>	Configures SNMP to use the selected VRF to communicate with the host receiver. The <i>ip-address</i> can be an IPv4 or IPv6 address. The VRF name can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. This command adds an entry into the ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB. Note By default, SNMP sends the traps using the management VRF. If you do not want to use the management VRF, you must use this command to specify the desired VRF.
Step 5	(Optional) show snmp host Example: <pre>switch(config)# show snmp host</pre>	Displays information about configured SNMP hosts.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Enable SNMP Notifications

You can enable or disable notifications. If you do not specify a notification name, Cisco NX-OS enables all notifications except BGP, EIGRP, and OSPF notifications.



Note The **snmp-server enable traps** command enables both traps and informs, depending on the configured notification host receivers.

The following table lists the commands that enable the notifications for Cisco NX-OS MIBs.

Table 16: Enabling SNMP Notifications

MIB	Related Commands
All notifications (except BGP, EIGRP, and OSPF)	<code>snmp-server enable traps</code>
CISCO-AAA-SERVER-MIB	<code>snmp-server enable traps aaa</code> <code>snmp-server enable traps aaa server-state-change</code>
CISCO-BGP4-MIB	<code>snmp-server enable traps bgp</code>
CISCO-CALLHOME-MIB	<code>snmp-server enable traps callhome</code> <code>snmp-server enable traps callhome event-notify</code> <code>snmp-server enable traps callhome smtp-send-fail</code>
CISCO-CONFIG-MAN-MIB	<code>snmp-server enable traps config</code> <code>snmp-server enable traps config ccmCLIRunningConfigChanged</code>
CISCO-EIGRP-MIB	<code>snmp-server enable traps eigrp [tag]</code>
CISCO-ERR-DISABLE-MIB	<code>snmp-server enable traps link cerrDisableInterfaceEventRev1</code>
ENTITY-MIB, CISCO-ENTITY-SENSOR-MIB	<code>snmp-server enable traps entity</code> <code>snmp-server enable traps entity entity_fan_status_change</code> <code>snmp-server enable traps entity entity_mib_change</code> <code>snmp-server enable traps entity entity_module_inserted</code> <code>snmp-server enable traps entity entity_module_removed</code> <code>snmp-server enable traps entity entity_module_status_change</code> <code>snmp-server enable traps entity entity_power_out_change</code> <code>snmp-server enable traps entity entity_power_status_change</code> <code>snmp-server enable traps entity entity_unrecognised_module</code>
CISCO-FEATURE-CONTROL-MIB	<code>snmp-server enable traps feature-control</code> <code>snmp-server enable traps feature-control FeatureOpStatusChange</code>

MIB	Related Commands
CISCO-HSRP-MIB	snmp-server enable traps hsrp snmp-server enable traps hsrp state-change
IF-MIB	snmp-server enable traps link snmp-server enable traps link extended-linkDown snmp-server enable traps link extended-linkUp snmp-server enable traps link cieLinkDown snmp-server enable traps link cieLinkUp snmp-server enable traps link linkDown snmp-server enable traps link linkUp
OSPF-MIB, OSPF-TRAP-MIB	snmp-server enable traps ospf [<i>tag</i>] snmp-server enable traps ospf lsa snmp-server enable traps ospf rate-limit <i>rate</i>
CISCO-RF-MIB	snmp-server enable traps rf snmp-server enable traps rf redundancy_framework
CISCO-RMON-MIB	snmp-server enable traps rmon snmp-server enable traps rmon fallingAlarm snmp-server enable traps rmon hcFallingAlarm snmp-server enable traps rmon hcRisingAlarm snmp-server enable traps rmon risingAlarm
SNMPv2-MIB	snmp-server enable traps snmp snmp-server enable traps snmp authentication
CISCO-MAC-NOTIFICATION-MIB	snmp-server enable trap link cmn-mac-move-notification
CISCO-PORT-STORM-CONTROL-MIB	storm-control action trap
CISCO-STP-EXTENSIONS-MIB	snmp-server enable traps stpx stpxMstInconsistencyUpdate
CISCO-STP-BRIDGE-MIB	snmp-server enable traps bridge snmp-server enable traps bridge newroot snmp-server enable traps bridge topologychange

MIB	Related Commands
CISCO-STPX-MIB	snmp-server enable traps stpx snmp-server enable traps stpx inconsistency snmp-server enable traps stpx loop-inconsistency snmp-server enable traps stpx root-inconsistency
CISCO-SYSTEM-EXT-MIB	snmp-server enable traps sysmgr snmp-server enable traps sysmgr cseFailSwCoreNotifyExtended
UPGRADE-MIB	snmp-server enable traps upgrade snmp-server enable traps upgrade UpgradeJobStatusNotify snmp-server enable traps upgrade UpgradeOpNotifyOnCompletion
VTP-MIB	snmp-server enable traps vtp snmp-server enable traps vtp notif snmp-server enable traps vtp vlancreate snmp-server enable traps vtp vlandelete

Use the following commands in the configuration mode shown to enable the specified notification:

Command	Purpose
snmp-server enable traps Example: <pre>switch(config)# snmp-server enable traps</pre>	Enables all SNMP notifications.
snmp-server enable traps aaa [server-state-change] Example: <pre>switch(config)# snmp-server enable traps aaa</pre>	Enables the AAA SNMP notifications. Optionally, enables the following specific notifications: • server-state-change—Enables AAA server state-change notifications.
snmp-server enable traps bgp Example: <pre>switch(config)# snmp-server enable traps bgp</pre>	Enables Border Gateway Protocol (BGP) SNMP notifications.
snmp-server enable traps bridge [newroot] [topologychange] Example: <pre>switch(config)# snmp-server enable traps bridge</pre>	Enables STP bridge SNMP notifications. Optionally, enables the following specific notifications: • newroot—Enables STP new root bridge notifications. • topologychange—Enables STP bridge topology-change notifications.

Command	Purpose
snmp-server enable traps callhome [event-notify] [smtp-send-fail] Example: <pre>switch(config)# snmp-server enable traps callhome</pre>	Enables Call Home notifications. Optionally, enables the following specific notifications: • event-notify—Enables Call Home external event notifications. • smtp-send-fail—Enables Simple Mail Transfer Protocol (SMTP) message send fail notifications.
snmp-server enable traps config [ccmCLIRunningConfigChanged] Example: <pre>switch(config)# snmp-server enable traps config</pre>	Enables SNMP notifications for configuration changes. • ccmCLIRunningConfigChanged—Enables SNMP notifications for configuration changes in the running or startup configuration.
snmp-server enable traps eigrp [tag] Example: <pre>switch(config)# snmp-server enable traps eigrp</pre>	Enables CISCO-EIGRP-MIB SNMP notifications.
snmp-server enable traps entity [entity_fan_status_change] [entity_mib_change] [entity_module_inserted] [entity_module_removed] [entity_module_status_change] [entity_power_out_change] [entity_power_status_change] [entity_unrecognised_module] Example: <pre>switch(config)# snmp-server enable traps entity</pre>	Enables ENTITY-MIB SNMP notifications. Optionally, enables the following specific notifications: • entity_fan_status_change—Enables entity fan status-change notifications. • entity_mib_change—Enables entity MIB change notifications. • entity_module_inserted—Enables entity module inserted notifications. • entity_module_removed—Enables entity module removed notifications. • entity_module_status_change—Enables entity module status-change notifications. • entity_power_out_change—Enables entity power-out change notifications. • entity_power_status_change—Enables entity power status-change notifications. • entity_unrecognised_module—Enables entity unrecognized module notifications.

Command	Purpose
snmp-server enable traps feature-control [FeatureOpStatusChange] Example: <pre>switch(config)# snmp-server enable traps feature-control</pre>	Enables feature-control SNMP notifications. Optionally, enables the following specific notifications: • FeatureOpStatusChange—Enables feature operation status-change notifications.
snmp-server enable traps hsrp state-change Example: <pre>switch(config)# snmp-server enable traps hsrp</pre>	Enables CISCO-HSRP-MIB SNMP notifications. Optionally, enables the following specific notifications: • state-change—Enables HSRP state-change notifications.
snmp-server enable traps license [notify-license-expiry] [notify-license-expiry-warning] [notify-licensefile-missing] [notify-no-license-for-feature] Example: <pre>switch(config)# snmp-server enable traps license</pre>	Enables ENTITY-MIB SNMP notifications. Optionally, enables the following specific notifications: • notify-license-expiry—Enables license expiry notifications. • notify-license-expiry-warning—Enables license expiry warning notifications. • notify-licensefile-missing—Enables license file-missing notifications. • notify-no-license-for-feature—Enables no-license-installed-for-feature notifications.

Command	Purpose
snmp-server enable traps link [cieLinkDown] [cieLinkUp] [cmn-mac-move-notification] [IETF-extended-linkDown] [IETF-extended-linkUp] [cisco-extended-linkDown] [cisco-extended-linkUp][linkDown] [linkUp] Example: <pre>switch(config)# snmp-server enable traps link</pre>	Enables IF-MIB link notifications. Optionally, enable the following specific notifications: • IETF-extended-linkDown—Enables Cisco extended link state down notifications. • IETF-extended-linkUp—Enables Cisco extended link state up notifications. • cmn-mac-move-notification—Enables MAC address move notifications. • cisco-extended-linkDown—Enables Internet Engineering Task Force (IETF) extended link state down notifications. • cisco-extended-linkUp—Enables Internet Engineering Task Force (IETF) extended link state up notifications. • linkDown—Enables IETF link state down notifications. • linkUp—Enables IETF link state up notifications.
snmp-server enable traps ospf [tag] [lsa] Example: <pre>switch(config)# snmp-server enable traps ospf</pre>	Enables Open Shortest Path First (OSPF) notifications. Optionally, enables the following specific notifications: • lsa—Enables OSPF link state advertisement (LSA) notifications.
snmp-server enable traps rf [redundancy-framework] Example: <pre>switch(config)# snmp-server enable traps rf</pre>	Enables redundancy framework (RF) SNMP notifications. Optionally, enables the following specific notifications: • redundancy-framework—Enables RF supervisor switchover MIB notifications.

Command	Purpose
snmp-server enable traps rmon [fallingAlarm] [hcFallingAlarm] [hcRisingAlarm] [risingAlarm] Example: <pre>switch(config)# snmp-server enable traps rmon</pre>	Enables remote monitoring (RMON) SNMP notifications. Optionally, enables the following specific notifications: • fallingAlarm—Enables RMON falling alarm notifications. • hcFallingAlarm—Enables RMON high-capacity falling alarm notifications. • hcRisingAlarm—Enables RMON high-capacity rising alarm notifications. • risingAlarm—Enables RMON rising alarm notifications.
snmp-server enable traps snmp [authentication] Example: <pre>switch(config)# snmp-server enable traps snmp</pre>	Enables general SNMP notifications. Optionally, enables the following specific notifications: • authentication—Enables SNMP authentication notifications.
snmp-server enable traps stpx [inconsistency] [loop-inconsistency] [root-inconsistency] Example: <pre>switch(config)# snmp-server enable traps stpx</pre>	Enables SNMP STPX notifications. Optionally, enables the following specific notifications: • inconsistency—Enables SNMP STPX MIB inconsistency update notifications. • loop-inconsistency—Enables SNMP STPX MIB loop-inconsistency update notifications. • root-inconsistency—Enables SNMP STPX MIB root-inconsistency update notifications.
snmp-server enable traps syslog [message-generated] Example: <pre>switch(config)# snmp-server enable traps syslog</pre>	Sends syslog messages as traps to the defined SNMP host. Optionally, enables the following specific notifications: • message-generated—Enables software log message generated notifications.
snmp-server enable traps sysmgr [cseFailSwCoreNotifyExtended] Example: <pre>switch(config)# snmp-server enable traps sysmgr</pre>	Enables software change notifications. Optionally, enables the following specific notifications: • cseFailSwCoreNotifyExtended—Enables software core notifications.

Command	Purpose
snmp-server enable traps upgrade [UpgradeJobStatusNotify] [UpgradeOpNotifyOnCompletion] Example: <pre>switch(config)# snmp-server enable traps upgrade</pre>	Enables upgrade notifications. Optionally, enables the following specific notifications: • UpgradeJobStatusNotify—Enables upgrade job status notifications. • UpgradeOpNotifyOnCompletion—Enables upgrade global status notifications.
snmp-server enable traps vtp [notifs] [vlancreate] [vlandelete] Example: <pre>switch(config)# snmp-server enable traps vtp</pre>	Enables VTP notifications. Optionally, enables the following specific notifications: • notifs—Enables VTP notifications. • vlancreate—Enables VLAN creation notifications. • vlandelete—Enables VLAN deletion notifications.
storm-control action traps Example: <pre>switch(config-if)# storm-control action traps</pre>	Enables traffic storm control notifications when the traffic storm control limit is reached.

Disable Link Notifications on an Interface

You can disable linkUp and linkDown notifications on an individual interface. You can use this limit notifications on flapping interface (an interface that transitions between up and down repeatedly).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface type slot/port Example: <pre>switch(config)# interface ethernet 2/2</pre>	Disables SNMP link-state traps for the interface. This command is enabled by default.
Step 3	no snmp trap link-status Example: <pre>switch(config-if)# no snmp trap link-status</pre>	Disables SNMP link-state traps for the interface. This command is enabled by default.

	Command or Action	Purpose
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Display SNMP ifIndex for an Interface

The SNMP ifIndex is used across multiple SNMP MIBs to link related interface information.

Procedure

	Command or Action	Purpose
Step 1	show interface snmp-ifindex Example: <pre>switch# show interface snmp-ifindex grep -i Eth12/1 Eth12/1 441974784 (0x1a580000)</pre>	Displays the persistent SNMP ifIndex value from the IF-MIB for all interfaces. Optionally, use the keyword and the grep keyword to search for a particular interface in the output.

Enable One-Time Authentication for SNMP over TCP

You can enable a one-time authentication for SNMP over a TCP session.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server tcp-session [auth] Example: <pre>switch(config)# snmp-server tcp-session</pre>	Enables a one-time authentication for SNMP over a TCP session. The default is disabled.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Assign SNMP Device Contact and Location Information

You can assign the device contact information, which is limited to 32 characters (without spaces) and the device location.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server contact <i>name</i> Example: <pre>switch(config)# snmp-server contact Admin</pre>	Configures sysContact, which is the SNMP contact name.
Step 3	snmp-server location <i>name</i> Example: <pre>switch(config)# snmp-server location Lab-7</pre>	Configures sysLocation, which is the SNMP location.
Step 4	(Optional) show snmp Example: <pre>switch(config)# show snmp</pre>	Displays information about one or more destination profiles.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure Context to Network Entity Mapping

You can configure an SNMP context to map to a logical network entity, such as a protocol instance or VRF.

Before you begin

Determine the logical network entity instance. For more information on VRFs and protocol instances, see the [Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide](#) or the [Cisco Nexus 9000 Series NX-OS Multicast Routing Configuration Guide](#).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] snmp-server context <i>context-name</i> [instance <i>instance-name</i>] [vrf <i>vrf-name</i>] [topology <i>topology-name</i>] Example: switch(config)# snmp-server context public1 vrf red	Maps an SNMP context to a protocol instance, VRF, or topology. The names can be any alphanumeric string up to 32 characters. The no option deletes the mapping between an SNMP context and a protocol instance, VRF, or topology. Note Do not enter an instance, VRF, or topology to delete a context mapping. If you use the instance, VRF, or topology keywords, you configure a mapping between the context and a zero-length string.
Step 3	(Optional) snmp-server mib community-map <i>community-name</i> context <i>context-name</i> Example: switch(config)# snmp-server mib community-map public context public1	Maps an SNMPv2c community to an SNMP context. The names can be any alphanumeric string up to 32 characters.
Step 4	(Optional) show snmp context Example: switch(config)# show snmp context	Displays information about one or more SNMP contexts.
Step 5	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Disable SNMP

You can disable SNMP on the device.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	no snmp-server protocol enable Example: switch(config)# no snmp-server protocol enable	Disables SNMP. SNMP is enabled by default. Note You cannot disable SNMPv1 without disabling SNMPv2. If you want to disable SNMPv1, then configure only SNMPv3, or disable SNMP entirely.

Modify SNMP Server Counter Cache Update Timer

You can modify how long, in seconds Cisco NX-OS holds the cache port state.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	snmp-server counter cache timeout seconds Example: switch(config)# snmp-server counter cache timeout 1200	Defines how long in seconds, the port states are held in the local cache. The counter cache is enabled by default, and the default cache timeout value is 10 seconds. When disabled, the default cache timeout value is 50 seconds. The range is 1-3600. Note For end of row (EoR) switching - The range is from 10 to 3600.
Step 3	(Optional) show running-config snmp all i cac Example: switch(config)# copy running-config snmp all i cac	Displays the configured SNMP-server counter cache update timeout value.
Step 4	no snmp-server counter cache enable Example: switch(config)# no snmp-server counter cache enable	Disables the counter cache update. Note When the counter cache update is disabled, the value set in the timeout parameter determines length of time the port states are held the counter cache.

Modify AAA Synchronization Time

You can modify how long Cisco NX-OS holds the synchronized user configuration.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server aaa-user cache-timeout seconds Example: <pre>switch(config)# snmp-server aaa-user cache-timeout 1200</pre>	Configures how long the AAA synchronized user configuration stays in the local cache. The range is from 1 to 86400 seconds. The default is 3600.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure SNMP Local Engine ID

Beginning with Cisco NX-OS Release 7.0(3)I6(1), you can configure the engine ID on a local device.



Note After you configure the SNMP local engine ID, you must reconfigure all SNMP users, any host configured with the V3 users, and the community strings. Beginning with Cisco NX-OS Release 7.0(3)I7(1), you need to reconfigure only the SNMP users and community strings.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	snmp-server engineID local engineid-string Example: <pre>switch(config)# snmp-server engineID local AA:BB:CC:1A:2C:10</pre>	Changes the SNMP engine ID of the local device. The local engine ID should be configured as a list of colon-specified hexadecimal octets, where there are even number of hexadecimal

	Command or Action	Purpose
		characters that range from 10 to 64 and every two hexadecimal characters are separated by a colon. For example, 80:00:02:b8:04:61:62:63.
Step 3	show snmp engineID Example: switch(config)# show snmp engineID	Displays the identification of the configured SNMP engine.
Step 4	[no] snmp-server engineID local engineid-string Example: switch(config)# no snmp-server engineID local AA:BB:CC:1A:2C:10	Disables the local engine ID and the default auto-generated engine ID is configured.
Step 5	Required: copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verify SNMP Configuration

To display SNMP configuration information, perform one of the following tasks:

Command	Purpose
show interface snmp-ifindex	Displays the SNMP ifIndex value for all interfaces (from IF-MIB).
show running-config snmp [all]	Displays the SNMP running configuration. SNMP users brought into 10.1(1), from releases prior to 10.1(1), are displayed with the configured privacy protocol, AES-128 or DES. New users (Release 10.1(1) and later) are by default configured with AES-128 protocol. Beginning with 9.3(8) release, SNMPv3 users under show run will be represented in SALT format instead of hash.
show snmp	Displays the SNMP status.

Command	Purpose
show snmp community	Displays the SNMP community strings. Note If the name of the SNMP context in the snmp-server mib community-map command is more than 11 characters, the output of the show snmp community command is displayed in a vertical format instead of a tabular format.
show snmp context	Displays the SNMP context mapping.
show snmp engineID	Displays the SNMP engineID.
show snmp group	Displays SNMP roles.
show snmp host	Displays information about configured SNMP hosts.
show snmp session	Displays SNMP sessions.
show snmp source-interface	Displays information about configured source interfaces.
show snmp trap	Displays the SNMP notifications enabled or disabled.
show snmp user	Displays SNMPv3 users.

Configuration Examples

Configure Cisco NX-OS to send the Cisco linkUp or Down notifications to one notification host receiver using the Blue VRF and defines two SNMP users, Admin and NMS

```
configure terminal
snmp-server contact Admin@company.com
snmp-server user Admin auth sha abcd1234 priv abcdefgh
snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID
00:00:00:63:00:01:00:22:32:15:10:03
snmp-server host 192.0.2.1 informs version 3 auth NMS
snmp-server host 192.0.2.1 use-vrf Blue
snmp-server enable traps link cisco
```

Configure SNMP to send traps using an inband port configured at the host level

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server host 171.71.48.164 version 2c public
switch(config)# snmp-server host 171.71.48.164 source-interface ethernet 1/2
```

```

switch(config)# show snmp host
-----
Host Port Version Level Type SecName
-----
171.71.48.164 162 v2c noauth trap public
Source interface: Ethernet 1/2
-----

switch(config)# snmp-server host 171.71.48.164 use-vrf default
switch(config)# show snmp host
-----
Host Port Version Level Type SecName
-----
171.71.48.164 162 v2c noauth trap public
Use VRF: default
Source interface: Ethernet 1/2
-----

```

Configure SNMP to send traps using a globally configured inband port

```

switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server source-interface traps ethernet 1/2
switch(config)# show snmp source-interface
-----
Notification source-interface
-----
trap Ethernet1/2
inform -
-----

switch(config)# snmp-server host 171.71.48.164 use_vrf default
switch(config)# show snmp host
-----
Host Port Version Level Type SecName
-----
171.71.48.164 162 v2c noauth trap public
Use VRF: default
Source interface: Ethernet 1/2
-----

```

Map VRF red to the SNMPv2c public community string

```

switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# vrf context red
switch(config-vrf)# exit
switch(config)# snmp-server context public1 vrf red
switch(config)# snmp-server mib community-map public context public1

```

Map OSPF instance Enterprise to the same SNMPv2c public community string

```

switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# feature ospf
switch(config)# router ospf Enterprise
switch(config-router)# exit
switch(config)# snmp-server context public1 instance Enterprise
switch(config)# snmp-server mib community-map public context public1

```

Additional References

Related Documents

Related Topic	Document Title
IP ACLs and AAA	<i>Cisco Nexus 9000 Series NX-OS Security Configuration Guide</i>
MIBs	<i>Cisco Nexus 7000 Series and 9000 Series NX-OS MIB Quick Reference</i>
RFC 3414	<i>User-based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMPv3)</i>
RFC 3415	<i>View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)</i>
MIBs related to SNMP	To locate and download supported MIBs, go to the following URL: https://cisco.github.io/cisco-mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

RFCs

RFC	Title
RFC 3414	<i>User-based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMPv3)</i>
RFC 3415	<i>View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)</i>

MIBs

MIBs	MIBs Link
MIBs related to SNMP	To locate and download supported MIBs, go to the following URL: https://cisco.github.io/cisco-mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html



CHAPTER 16

Configuring RMON

This chapter describes how to configure the remote monitoring (RMON) feature on Cisco NX-OS devices.

This chapter contains the following sections:

- [About RMON, on page 275](#)
- [Guidelines and Limitations for RMON, on page 277](#)
- [Default Settings for RMON, on page 277](#)
- [Configuring RMON, on page 277](#)
- [Verifying the RMON Configuration, on page 279](#)
- [Configuration Examples for RMON, on page 280](#)
- [Additional References, on page 280](#)

About RMON

RMON is a Simple Network Management Protocol (SNMP) Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. Cisco NX-OS supports RMON alarms, events, and logs to monitor Cisco NX-OS devices.

An RMON alarm monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified threshold value (threshold), and resets the alarm at another threshold value. You can use alarms with RMON events to generate a log entry or an SNMP notification when the RMON alarm triggers.

RMON is enabled by default, but no alarms are configured in Cisco NX-OS. You can configure RMON alarms by using the CLI or an SNMP-compatible network management station.

RMON Alarms

You can set an alarm on any MIB object that resolves into an SNMP INTEGER type. The specified object must be an existing SNMP MIB object in standard dot notation (for example, 1.3.6.1.2.1.2.2.1.14 represents ifInOctets.14).

When you create an alarm, you specify the following parameters:

- MIB object to monitor.
- Sampling interval—The interval that the device uses to collect a sample value of the MIB object.

- Sample type—Absolute samples take the current snapshot of the MIB object value. Delta samples take two consecutive samples and calculate the difference between them.
- Rising threshold—The value at which the device triggers a rising alarm or resets a falling alarm.
- Falling threshold—The value at which the device triggers a falling alarm or resets a rising alarm.
- Events—The action that the device takes when an alarm (rising or falling) triggers.



Note Use the `hcalarms` option to set an alarm on a 64-bit integer MIB object.

For example, you can set a delta type rising alarm on an error counter MIB object. If the error counter delta exceeds this value, you can trigger an event that sends an SNMP notification and logs the rising alarm event. This rising alarm will not occur again until the delta sample for the error counter drops below the falling threshold.



Note The falling threshold must be less than the rising threshold.

RMON Events

You can associate a particular event to each RMON alarm. RMON supports the following event types:

- SNMP notification—Sends an SNMP risingAlarm or fallingAlarm notification when the associated alarm triggers.
- Log—Adds an entry in the RMON log table when the associated alarm triggers.
- Both—Sends an SNMP notification and adds an entry in the RMON log table when the associated alarm triggers.

You can specify a different even for a falling alarm and a rising alarm.



Note You may choose to use the default RMON events template configuration or you can delete these entries and create new RMON events. Until you create RMON alarm configurations, no alarms will be triggered by these configurations.

High Availability for RMON

Cisco NX-OS supports stateless restarts for RMON. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

Virtualization Support for RMON

Cisco NX-OS supports one instance of RMON.

RMON is virtual routing and forwarding (VRF) aware. You can configure RMON to use a particular VRF to reach the RMON SMTP server.

Guidelines and Limitations for RMON

RMON has the following configuration guidelines and limitations:

- You must configure an SNMP user and a notification receiver to use the SNMP notification event type.
- You can configure an RMON alarm only on a MIB object that resolves to an integer.
- When you configure an RMON alarm, the object identifier must be complete with its index so that it refers to only one object. For example, 1.3.6.1.2.1.2.2.1.14 corresponds to `cpmCPUTotal5minRev`, and .1 corresponds to index `cpmCPUTotalIndex`, which creates object identifier 1.3.6.1.2.1.2.2.1.14.1.

Default Settings for RMON

The following table lists the default settings for RMON parameters.

Parameters	Default
RMON	Enabled
Alarms	None configured
Events	Configured (but triggered event causes nothing)

Configuring RMON



Note Be aware that the Cisco NX-OS commands for this feature may differ from those commands used in Cisco IOS.

Configuring RMON Alarms

You can configure RMON alarms on any integer-based SNMP MIB object.

You can optionally specify the following parameters:

- The event number to trigger if the rising or falling threshold exceeds the specified limit.
- The owner of the alarm.

Ensure you have configured an SNMP user and enabled SNMP notifications.

Before you begin

Make sure that you have configured an SNMP user and enabled SNMP notifications.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	rmon alarm <i>index mib-object sample-interval</i> {absolute delta} rising-threshold <i>value</i> [event-index] falling-threshold <i>value</i> [event-index] [owner name] Example: <pre>switch(config)# rmon alarm 20 1.3.6.1.2.1.2.2.1.14.1 2900 delta rising-threshold 1500 1 falling-threshold 0 owner test</pre>	Creates an RMON alarm. The value range is from –2147483647 to 2147483647. The owner name can be any alphanumeric string.
Step 3	rmon hcalarm <i>index mib-object sample-interval</i> {absolute delta} rising-threshold-high <i>value</i> rising-threshold-low <i>value</i> [event-index] falling-threshold-high <i>value</i> falling-threshold-low <i>value</i> [event-index] [owner name] [storagetype type] Example: <pre>switch(config)# rmon alarm 20 1.3.6.1.2.1.2.2.1.14.16777216 2900 delta rising-threshold-high 15 rising-threshold-low 151 falling-threshold-high 0 falling-threshold-low 0 owner test</pre>	Creates an RMON high-capacity alarm. The value range is from –2147483647 to 2147483647. The owner name can be any alphanumeric string. The storage type range is from 1 to 5.
Step 4	(Optional) show rmon {alarms hcalarms} Example: <pre>switch(config)# show rmon alarms</pre>	Displays information about RMON alarms or high-capacity alarms.
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring RMON Events

You can configure RMON events to associate with RMON alarms. You can reuse the same event with multiple RMON alarms.

Before you begin

Make sure you have configured an SNMP user and enabled SNMP notifications.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	rmon event <i>index</i> [description string] [log] [trap string] [owner name] Example: switch(config)# rmon event 1 trap trap1	Configures an RMON event. The description string, trap string, and owner name can be any alphanumeric string.
Step 3	(Optional) show rmon events Example: switch(config)# show rmon events	Displays information about RMON events.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Verifying the RMON Configuration

To display RMON configuration information, perform one of the following tasks:

Command	Purpose
show rmon alarms	Displays information about RMON alarms.
show rmon events	Displays information about RMON events.
show rmon hcalarms	Displays information about RMON high-capacity alarms.
show rmon logs	Displays information about RMON logs.

Configuration Examples for RMON

This example shows how to create a delta rising alarm on ifInOctets.14 and associates a notification event with this alarm:

```
configure terminal
rmon alarm 20 1.3.6.1.2.1.2.2.1.14.1 2900 delta rising-threshold 1500 1 falling-threshold
0 owner test
rmon event 1 trap trap1
```

Additional References

MIBs

MIBs	MIBs Link
MIBs related to RMON	To locate and download supported MIBs, go to the following https://cisco.github.io/cisco-mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html



CHAPTER 17

Configuring Online Diagnostics

This chapter describes how to configure the generic online diagnostics (GOLD) feature on Cisco NX-OS devices.

- [About Online Diagnostics, on page 281](#)
- [Guidelines and Limitations for Online Diagnostics, on page 288](#)
- [Default Settings for Online Diagnostics, on page 289](#)
- [Configuring Online Diagnostics, on page 289](#)
- [Verifying the Online Diagnostics Configuration, on page 293](#)
- [Configuration Examples for Online Diagnostics, on page 294](#)

About 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 tests that check different hardware components and verify the data path and control signals. Disruptive online diagnostic tests (such as the disruptive loopback test) and nondisruptive online diagnostic tests (such as the ASIC register check) run during bootup, line module online insertion and removal (OIR), and system reset. The nondisruptive online diagnostic tests run as part of the background health monitoring, and you can run these tests on demand.

Online diagnostics are categorized as bootup, runtime or health-monitoring diagnostics, and on-demand diagnostics. Bootup diagnostics run during bootup, health-monitoring tests run in the background, and on-demand diagnostics run once or at user-designated intervals when the device is connected to a live network.

Bootup Diagnostics

Bootup diagnostics run during bootup and detect faulty hardware before Cisco NX-OS brings a module online. For example, if you insert a faulty module in the device, bootup diagnostics test the module and take it offline before the device uses the module to forward traffic.

Bootup diagnostics also check the connectivity between the supervisor and module hardware and the data and control paths for all the ASICs. The following table describes the bootup diagnostic tests for a module and a supervisor.

Table 17: Bootup Diagnostics

Diagnostic	Description
OBFL	Verifies the integrity of the onboard failure logging (OBFL) flash.
MacSecPortLoopback (Cisco Nexus 9736C-FX and 9736Q-FX line cards only)	Tests the packet path from Supervisor to each physical front panel port on the ASIC, the MACSEC capabilities of each port, and the Encryption and Decryption capabilities of the Cisco Nexus 9736C-FX and 9736Q-FX line cards. The MacSecPortLoopback test runs at boot time when the diagnostic bootup level is set to complete. The MacSecPortLoopback test runs on every port of the 36 front ports on the Cisco Nexus 9736C-FX and 9736Q-FX line cards, including ports that are broken out. The MAC sec hardware is tested for the four available cipher suite algorithms: GCM-AES-128, GCM-AES-256, GCM-AES-XPB-128, and GCM-AES-XPB-256. Note If a MacSecPortLoopback test failure occurs, the test reports in the form of SYSLOG or OBFL. When a test failure occurs, the port is taken down and display <code>MACsec failure</code> in the show interface CLI output. You can skip the MACsec test by setting the diagnostic bootup level to either minimal or bypass.
USB	Nondisruptive test. Checks the USB controller initialization on a module.
ManagementPortLoopback	Disruptive test, not an on-demand test. Tests loopback on the management port of a module.
EOBCPortLoopback	Disruptive test, not an on-demand test. Ethernet out of band.

Bootup diagnostics log failures to onboard failure logging (OBFL) and syslog and trigger a diagnostic LED indication (on, off, pass, or fail).

You can configure the device to either bypass the bootup diagnostics or run the complete set of bootup diagnostics.

Runtime or Health Monitoring Diagnostics

Runtime diagnostics are also called health monitoring (HM) diagnostics. These diagnostics provide information about the health of a live device. They detect runtime hardware errors, memory errors, the degradation of hardware modules over time, software faults, and resource exhaustion.

Health monitoring diagnostics are non-disruptive and run in the background to ensure the health of a device that is processing live network traffic. You can enable or disable health monitoring tests or change their runtime interval.

The following table describes the health monitoring diagnostics and test IDs for a module and a supervisor.



Note Some tests may or may not be present, depending on the capabilities of the module. A list of tests available to the module can be found using the CLI command **show diagnostic content module <module>**.

Table 18: Health Monitoring Non-disruptive Diagnostics

Diagnostic	Default Interval	Default Setting	Description	Corrective Action
Module				
ACT2	30 minutes	active	Verifies the integrity of the security device on the module.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "ACT2" test
ASICRegisterCheck	modular switches: 1 minute non-modular switches: 20 seconds and a minimum configuration default simulation interval of 10 seconds	active	Validates read/write access to the ASICs on a module.	Do CallHome, log error, and disable further HM testing for that ASIC device/instance after 20 consecutive failures of GOLD "ASICRegisterCheck" test
PrimaryBootROM	24 hours 1	active	Verifies the integrity of the primary boot device on a module.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "PrimaryBootROM" test
SecondaryBootROM	24 hours 1	active	Verifies the integrity of the secondary boot device on a module.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "SecondaryBootROM" test
BootupPortLoopback	Only on bootup	Only on boot up - active	Checks if the supervisor to front-panel port (and back) path is operational. For every front port, the test generates a packet on an active supervisor, sends the packet toward a target port, and, using the internal loopback inside a front port, redirects the packet back to the active supervisor.	Do CallHome, Error-disable affected ports, log error testing on affected ports after 1 consecutive failures of GOLD "BootupPortLoopback" test

Diagnostic	Default Interval	Default Setting	Description	Corrective Action
PortLoopback	30 minutes	active	Checks diagnostics on a per-port basis on all admin down ports.	Do CallHome, log error in Syslog/OBFL/Exception Log, and disable further HM testing on affected ports after 10 consecutive failures of GOLD "PortLoopback" test
RewriteEngineLoopback	1 minute	active	Verifies the integrity of the nondisruptive loopback for all ports up to the 1 Engine ASIC device.	Do CallHome, log error in Syslog/OBFL/Exception Log, and disable further HM testing on affected ports after 10 consecutive failures of GOLD "RewriteEngine" test
AsicMemory	Only on boot up	Only on boot up - inactive	Checks if the AsicMemory is consistent using the Mbist bit in the ASIC.	Do CallHome and log error when GOLD "AsicMemory" test fails. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. Note To avoid a kernel panic when the test fails, you can override the EEM system policy.
FpgaRegTest	30 seconds	Health monitoring test - every 30 seconds - active	Test the FPGA status by read/write to FPGA.	Do CallHome, log error, disable further HM testing after 20 consecutive failures of GOLD "FpgaRegTest" test. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. Note To avoid a kernel panic when the test fails, you can override the EEM system policy.

Diagnostic	Default Interval	Default Setting	Description	Corrective Action
L2ACLRedirect	1 minute	Health monitoring test - every minute - active	Checks if the active inband path is operational. The test generates a packet on an active supervisor through the active fabric module. It then sends the packet toward the front panel port (physical interface on the line card) and, using the ACL entry, redirects the packet back to the active supervisor.	Do CallHome, log error, disable further HM testing after 10 consecutive failures of L2ACLRedirect test. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. Note To avoid a kernel panic when the test fails, you can override the EEM system policy.
OBFL	30 minutes	active	Verifies the integrity of the onboard failure logging (OBFL) flash, and monitors for available storage in the device.	
FabricConnectivityTest	1 minute	active	Verifies fabric/linecard link status. Validates that the fabric links are functioning. Note Only available on Cisco Nexus 9500-R series line cards and Cisco N9K-X9836DM-A line cards.	

Diagnostic	Default Interval	Default Setting	Description	Corrective Action
FabricReachabilityTest	1 minute	active	Verifies fabric/linecard reachability status. Validates that each fabric component has a valid path to every other fabric component in the system. Note Only available on Cisco Nexus 9500-R series line cards and Cisco N9K-X9836DM-A line cards.	
Supervisor				
Backplane	30 minutes	active	Verifies the integrity of the backplane SPROM devices.	
NVRAM	5 minutes	active	Verifies the sanity of the NVRAM blocks on a supervisor.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "NVRAM" test
RealTimeClock	5 minutes	active	Verifies that the real-time clock on the supervisor is ticking.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "RealTimeClock" test
PrimaryBootROM	30 minutes	active	Verifies the integrity of the primary boot device on the supervisor.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "PrimaryBootROM" test
SecondaryBootROM	30 minutes	active	Verifies the integrity of the secondary boot device on the supervisor.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "SecondaryBootROM" test
BootFlash	30 minutes	active	Verifies access to the bootflash devices.	Do CallHome and log error when GOLD "BootFlash" test fails

Diagnostic	Default Interval	Default Setting	Description	Corrective Action
USB	30 minutes	active	Verifies access to the USB devices.	Do Call Home and log error when GOLD "USB" test fails
SystemMgmtBus	30 seconds	active	Verifies the availability of the system management bus.	Do Call Home, log error, and disable further HM testing for that fan or power supply after 20 consecutive failures of GOLD "SystemMgmtBus" test
Mce	30 minutes	Health monitoring test - 30 minutes - active	This test uses the mcd_dameon and reports any machine check error reported by the Kernel.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "Mce" test
Pcie	Only on boot up	Only on boot up - inactive	Reads PCIe status registers and check for any error on the PCIe device.	Do CallHome and log error when GOLD "Pcie" test fails
Console	Only on boot up	Only on boot up - inactive	This runs a port loopback test on the management port on boot up to check for its consistency.	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "Console" test
FpgaRegTest	30 seconds	Health monitoring test - every 30 seconds - active	Test the FPGA status by read/write to FPGA. Note FpgaRegTest on Cisco Nexus 9808 switch for Fabric Modules (19-26) will be displayed under the FpgaRegTest result of Active-SUP.	Do CallHome, log error, disable further HM testing after 20 consecutive failures of GOLD "FpgaRegTest" test. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. Note To avoid a kernel panic when the test fails, you can override the EEM system policy.

¹ Minimum configurable test interval is 6 hours

On-Demand Diagnostics

On-demand tests help localize faults and are usually needed in one of the following situations:

- To respond to an event that has occurred, such as isolating a fault.
- In anticipation of an event that may occur, such as a resource exceeding its utilization limit.

You can run all the health monitoring tests on demand. You can schedule on-demand diagnostics to run immediately.

You can also modify the default interval for a health monitoring test.

High Availability

A key part of high availability is detecting hardware failures and taking corrective action while the device runs in a live network. Online diagnostics in high availability detect hardware failures and provide feedback to high availability software components to make switchover decisions.

Cisco NX-OS supports stateless restarts for online diagnostics. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

Virtualization Support

Online diagnostics are virtual routing and forwarding (VRF) aware. You can configure online diagnostics to use a particular VRF to reach the online diagnostics SMTP server.

Guidelines and Limitations for Online Diagnostics

Online diagnostics has the following configuration guidelines and limitations:

- The following Cisco Nexus platform switches and line cards do not support the run-time PortLoopback test but do support the BootupPortLoopback test:

Switches

- Cisco Nexus 92160YC-X
- Cisco Nexus 92304QC
- Cisco Nexus 9264PQ
- Cisco Nexus 9272Q
- Cisco Nexus 9232C
- Cisco Nexus 9236C
- Cisco Nexus 9256PV
- Cisco Nexus 92300YC
- Cisco Nexus 93108TC-EX
- Cisco Nexus 93108TC-EX-24
- Cisco Nexus 93180LC-EX
- Cisco Nexus 93180YC-EX
- Cisco Nexus 93180YC-EXU
- Cisco Nexus 93180YC-EX-24

- Cisco Nexus 9232E-B1
- Cisco Nexus 93180YC-FX3S

Line Cards

- Cisco Nexus 9736C-EX
 - Cisco Nexus 97160YC-EX
 - Cisco Nexus 9732C-EX
 - Cisco Nexus 9732C-EXM
-
- You cannot run disruptive online diagnostic tests on demand.
 - Interface Rx and Tx packet counters are incremented (approximately four packets every 15 minutes) for ports in the shutdown state.
 - The PortLoopback test is periodic, so the packet counter is incremented on admin down ports every 30 minutes. The test runs only on admin down ports. When a port is unshut, the counters are not affected.
 - When a port fails for the per-port BootupPortLoopback test, the port enters the error-disabled state. (To remove this state, enter the **shutdown** and **no shutdown** commands on the port.)
 - Beginning with Cisco NX-OS Release 10.3(1)F, Generic Online Diagnostics (GOLD) is supported on the Cisco Nexus 9800 platform switches.

Default Settings for Online Diagnostics

The following table lists the default settings for online diagnostic parameters.

Parameters	Default
Bootup diagnostics level	complete
Nondisruptive tests	active

Configuring Online Diagnostics



Note Be aware that the Cisco NX-OS commands for this feature may differ from those commands used in Cisco IOS.

Setting the Bootup Diagnostic Level

You can configure the bootup diagnostics to run the complete set of tests, or you can bypass all bootup diagnostic tests for a faster module bootup time.



Note We recommend that you set the bootup online diagnostics level to complete. We do not recommend bypassing the bootup online diagnostics.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	diagnostic bootup level {complete minimal bypass} Example: <pre>switch(config)# diagnostic bootup level complete</pre>	Configures the bootup diagnostic level to trigger diagnostics as follows when the device boots: • complete—Perform a complete set of bootup diagnostics. The default is complete. • minimal—Perform a minimal set of bootup diagnostics for the supervisor engine and bootup port loopback tests. • bypass—Do not perform any bootup diagnostics.
Step 3	(Optional) show diagnostic bootup level Example: <pre>switch(config)# show diagnostic bootup level</pre>	Displays the bootup diagnostic level (bypass or complete) that is currently in place on the device.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Activating a Diagnostic Test

You can set a diagnostic test as active and optionally modify the interval (in hours, minutes, and seconds) at which the test runs.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	diagnostic monitor interval module slot test <code>[test-id name all] hour hour min minute second second</code> Example: <pre>switch(config)# diagnostic monitor interval module 6 test 3 hour 1 min 0 second 0</pre>	Configures the interval at which the specified test is run. If no interval is set, the test runs at the interval set previously, or the default interval. The argument ranges are as follows: • <i>slot</i>—The range is from 1 to 10. • <i>test-id</i>—The range is from 1 to 14. • <i>name</i>—Can be any case-sensitive, alphanumeric string up to 32 characters. • <i>hour</i>—The range is from 0 to 23 hours. • <i>minute</i>—The range is from 0 to 59 minutes. • <i>second</i>—The range is from 0 to 59 seconds.
Step 3	[no] diagnostic monitor module slot test <code>[test-id name all]</code> Example: <pre>switch(config)# diagnostic monitor interval module 6 test 3</pre>	Activates the specified test. The argument ranges are as follows: • <i>slot</i>—The range is from 1 to 10. • <i>test-id</i>—The range is from 1 to 14. • <i>name</i>—Can be any case-sensitive, alphanumeric string up to 32 characters. The [no] form of this command inactivates the specified test. Inactive tests keep their current configuration but do not run at the scheduled interval.
Step 4	(Optional) show diagnostic content module <code>{slot all}</code> Example: <pre>switch(config)# show diagnostic content module 6</pre>	Displays information about the diagnostics and their attributes.

Starting or Stopping an On-Demand Diagnostic Test

You can start or stop an on-demand diagnostic test. You can optionally modify the number of iterations to repeat this test, and the action to take if the test fails.

We recommend that you only manually start a disruptive diagnostic test during a scheduled network maintenance time.

Procedure

	Command or Action	Purpose
Step 1	(Optional) diagnostic ondemand iteration <i>number</i> Example: switch# diagnostic ondemand iteration 5	Configures the number of times that the on-demand test runs. The range is from 1 to 999. The default is 1.
Step 2	(Optional) diagnostic ondemand action-on-failure { continue failure-count <i>num-fails</i> stop } Example: switch# diagnostic ondemand action-on-failure stop	Configures the action to take if the on-demand test fails. The <i>num-fails</i> range is from 1 to 999. The default is 1.
Step 3	Required: diagnostic start module slot test [<i>test-id</i> <i>name</i> all non-disruptive] [port <i>port-number</i> all] Example: switch# diagnostic start module 6 test all	Starts one or more diagnostic tests on a module. The module slot range is from 1 to 10. The <i>test-id</i> range is from 1 to 14. The test name can be any case-sensitive, alphanumeric string up to 32 characters. The port range is from 1 to 48.
Step 4	Required: diagnostic stop module slot test [<i>test-id</i> <i>name</i> all] Example: switch# diagnostic stop module 6 test all	Stops one or more diagnostic tests on a module. The module slot range is from 1 to 10. The <i>test-id</i> range is from 1 to 14. The test name can be any case-sensitive, alphanumeric string up to 32 characters.
Step 5	(Optional) show diagnostic status module slot Example: switch# show diagnostic status module 6	Verifies that the diagnostic has been scheduled.

Simulating Diagnostic Results

You can simulate a diagnostic test result.

Procedure

	Command or Action	Purpose
Step 1	diagnostic test simulation module slot test <i>test-id</i> { fail random-fail success } [port <i>number</i> all] Example: switch# diagnostic test simulation module 2 test 2 fail	Simulates a test result. The <i>test-id</i> range is from 1 to 14. The port range is from 1 to 48.

Clearing Diagnostic Results

You can clear diagnostic test results.

Procedure

	Command or Action	Purpose
Step 1	diagnostic clear result module [<i>slot</i> all] test { <i>test-id</i> all } Example: <pre>switch# diagnostic clear result module 2 test all</pre>	Clears the test result for the specified test. The argument ranges are as follows: • <i>slot</i>—The range is from 1 to 10. • <i>test-id</i>—The range is from 1 to 14.
Step 2	diagnostic test simulation module <i>slot</i> test <i>test-id</i> clear Example: <pre>switch# diagnostic test simulation module 2 test 2 clear</pre>	Clears the simulated test result. The <i>test-id</i> range is from 1 to 14.

Verifying the Online Diagnostics Configuration

To display online diagnostics configuration information, perform one of the following tasks:

Command	Purpose
show diagnostic bootup level	Displays information about bootup diagnostics.
show diagnostic content module { <i>slot</i> all }	Displays information about diagnostic test content for a module.
show diagnostic description module <i>slot</i> test [<i>test-name</i> all]	Displays the diagnostic description.
show diagnostic events [error info]	Displays diagnostic events by error and information event type.
show diagnostic ondemand setting	Displays information about on-demand diagnostics.
show diagnostic result module <i>slot</i> [test [<i>test-name</i> all]] [detail]	Displays information about the results of a diagnostic.
show diagnostic simulation module <i>slot</i>	Displays information about a simulated diagnostic.
show diagnostic status module <i>slot</i>	Displays the test status for all tests on a module.
show hardware capacity [eobc forwarding interface module power]	Displays information about the hardware capabilities and current hardware utilization by the system.
show module	Displays module information including the online diagnostic test status.

Configuration Examples for Online Diagnostics

This example shows how to start all on-demand tests on module 6:

```
diagnostic start module 6 test all
```

This example shows how to activate test 2 and set the test interval on module 6:

```
configure terminal
diagnostic monitor module 6 test 2
diagnostic monitor interval module 6 test 2 hour 3 min 30 sec 0
```



CHAPTER 18

Configuring the Embedded Event Manager

This chapter describes how to configure the Embedded Event Manager (EEM) to detect and handle critical events on Cisco NX-OS devices.

- [About EEM, on page 295](#)
- [Prerequisites for EEM, on page 299](#)
- [Guidelines and Limitations for EEM, on page 299](#)
- [Default Settings for EEM, on page 300](#)
- [Configuring EEM, on page 301](#)
- [Verifying the EEM Configuration, on page 314](#)
- [Configuration Examples for EEM, on page 315](#)
- [Event Log Auto-Collection and Backup, on page 316](#)

About EEM

EEM monitors events that occur on your device and takes action to recover or troubleshoot these events, based on your configuration.

EEM consists of three major components:

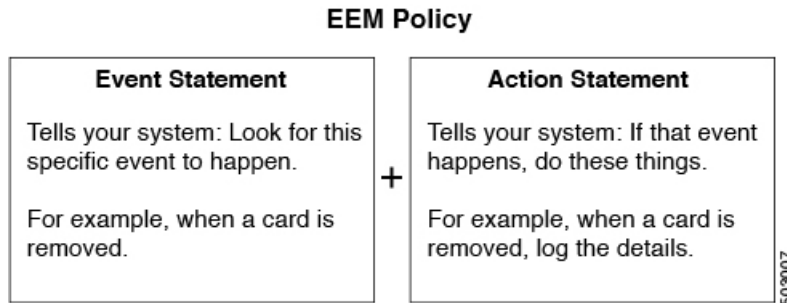
- Event statements—Events to monitor from another Cisco NX-OS component that may require some action, workaround, or notification.
- Action statements—An action that EEM can take, such as executing CLI commands, sending an email through the use of Smart Call Home feature, and disabling an interface to recover from an event.
- Policies—An event that is paired with one or more actions to troubleshoot or recover from the event.

Policies

An EEM policy consists of an event statement and one or more action statements. The event statement defines the event to look for as well as the filtering characteristics for the event. The action statement defines the action EEM takes when the event occurs.

This figure shows the two basic statements in an EEM policy.

Figure 5: EEM Policy Statements



You can configure EEM policies using the command-line interface (CLI) or a VSH script.

EEM gives you a device-wide view of policy management. You configure EEM policies on the supervisor, and EEM pushes the policy to the correct module based on the event type. EEM takes any actions for a triggered event either locally on the module or on the supervisor (the default option).

EEM maintains event logs on the supervisor.

Cisco NX-OS has a number of preconfigured system policies. These system policies define many common events and actions for the device. System policy names begin with two underscore characters (___).

You can create user policies to suit your network. If you create a user policy, any actions in your policy occur after EEM triggers any system policy actions that are related to the same event as your policy.

You can also override some system policies. The overrides that you configure take the place of the system policy. You can override the event or the actions.

Use the **show event manager system-policy** command to view the preconfigured system policies and determine which policies that you can override.



Note You should use the **show running-config eem** command to check the configuration of each policy. An override policy that consists of an event statement and no action statement triggers no action and no notification of failures.



Note Your override policy should always include an event statement. An override policy without an event statement overrides all possible events in the system policy.

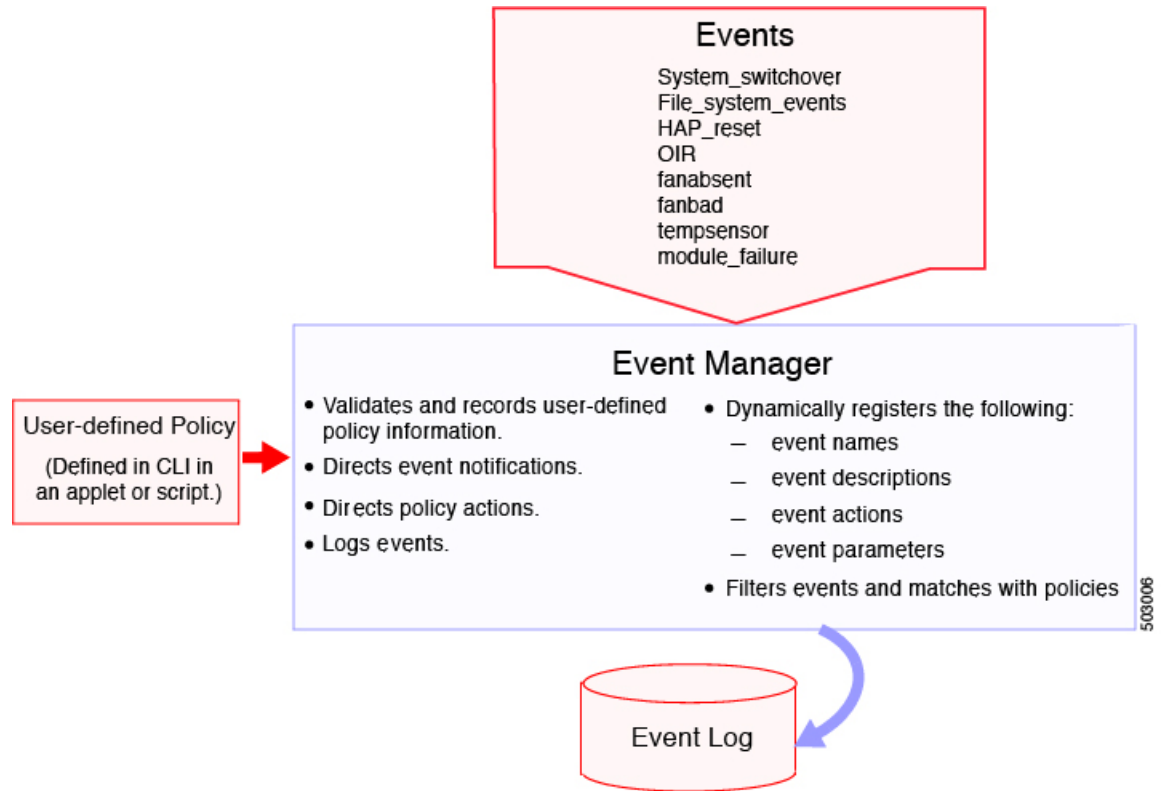
Event Statements

An event is any device activity for which some action, such as a workaround or a notification, should be taken. In many cases, these events are related to faults in the device such as when an interface or a fan malfunctions.

EEM defines event filters so only critical events or multiple occurrences of an event within a specified time period trigger an associated action.

This figure shows events that are handled by EEM.

Figure 6: EEM Overview



Event statements specify the event that triggers a policy to run. You can configure multiple event triggers.

EEM schedules and runs policies on the basis of event statements. EEM examines the event and action commands and runs them as defined.



Note If you want to allow the triggered event to process any default actions, you must configure the EEM policy to allow the event default action statement.

Action Statements

Action statements describe the action triggered by a policy. Each policy can have multiple action statements. If no action is associated with a policy, EEM still observes events but takes no actions.

EEM supports the following actions in action statements:

- Execute any CLI commands.
- Update a counter.
- Log an exception.
- Force the shutdown of any module.
- Reload the device.

- Shut down specified modules because the power is over budget.
- Generate a syslog message.
- Generate a Call Home event.
- Generate an SNMP notification.
- Use the default action for the system policy.



Note EEM can only process a complete action cli list of up to 1024 characters in total. If more actions are required, you must define them as a new redundant applet with same trigger.



Note If you want to allow the triggered event to process any default actions, you must configure the EEM policy to allow the default action. For example, if you match a CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM will not allow the CLI command to execute.



Note Verify that your action statements within your user policy or overriding policy do not negate each other or adversely affect the associated system policy.

VSH Script Policies

You can also write policies in a VSH script, using a text editor. These policies have an event statement and action statement(s) just as other policies, and these policies can either augment or override system policies. After you write your VSH script policy, copy it to the device and activate it.

Environment Variables

You can define environment variables for EEM that are available for all policies. Environment variables are useful for configuring common values that you can use in multiple policies. For example, you can create an environment variable for the IP address of an external email server.

You can use an environment variable in action statements by using the parameter substitution format.

This example shows a sample action statement to force a module 1 shutdown, with a reset reason of "EEM action."

```
switch (config-eem-policy)# action 1.0 forceshut module 1 reset-reason "EEM action."
```

If you define an environment variable for the shutdown reason, called default-reason, you can replace that reset reason with the environment variable, as shown in the following example.

```
switch (config-eem-policy)# action 1.0 foreshut module 1 reset-reason $default-reason
```

You can reuse this environment variable in any policy.

EEM Event Correlation

You can trigger an EEM policy based on a combination of events. First, you use the **tag** keyword to create and differentiate multiple events in the EEM policy. Then using a set of boolean operators (**and**, **or**, **andnot**), along with the count and time, you can define a combination of these events to trigger a custom action.

High Availability

Cisco NX-OS supports stateless restarts for EEM. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

Virtualization Support

Not all actions or events are visible. You must have network-admin privileges to configure policies.

Prerequisites for EEM

EEM has the following prerequisites:

- You must have network-admin user privileges to configure EEM.

Guidelines and Limitations for EEM

EEM has the following configuration guidelines and limitations:

- The maximum number of configurable EEM policies is 500.
- Action statements within your user policy or overriding policy should not negate each other or adversely affect the associated system policy.
- To allow a triggered event to process any default actions, you must configure the EEM policy to allow the default action. For example, if you match a CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM will not allow the CLI command to execute.
- Only 10 triggers from the same client (for example: vsd is the client for "event cli", snmp is the client for "event snmp" etc.) are allowed to be published within one second.
- Action with option **collect** must be always first action in the event applet statement.
- The following guidelines apply to Event Log Auto-Collection and Backup:
 - By default, enabled log collection on a switch provides between 15 minutes to several hours of event logs depending on size, scale and component activity.
 - To be able to collect relevant logs that span a longer period, only enable event log retention for the specific services/features you need. See "Enabling Extended Log File Retention For a Single Service". You can also export the internal event logs. See "External Log File Storage".
 - When troubleshooting, it is good practice to manually collect a snapshot of internal event logs in real time. See "Generating a Local Copy of Recent Log Files".

- When you configure an EEM policy action to collect **show tech** commands, make sure to allocate enough time for the **show tech** commands to complete before the same action is called again.
- Note the following about override policies:
 - An override policy that consists of an event statement without an action statement triggers no action and no notification of failures.
 - An override policy without an event statement overrides all possible events in the system policy.
- The following rules apply to regular command expressions:
 - All regular expressions must conform to the Portable Operating System Interface for uniX (POSIX) extended standard.
 - All keywords must be expanded.
 - Only the * symbol can be used for argument replacement.
- Note the following about EEM event correlation:
 - EEM event correlation is supported only on the supervisor module.
 - EEM event correlation is not supported across different modules within a single policy.
 - EEM event correlation supports up to four event statements in a single policy. The event types can be the same or different, but only these event types are supported: cli, counter, module, module-failure, oir, snmp, and syslog.
 - EEM event correlation does not override the system default policies.
- When more than one event statement is included in an EEM policy, each event statement must have a **tag** keyword with a unique tag argument.
- Default action execution is not supported for policies that are configured with tagged events.
- You can invoke EEM from Python. For more information about Python, see the [Cisco Nexus 9000 Series NX-OS Programmability Guide](#).
- Beginning with Cisco NX-OS Release 10.3(1)F, the default auto-collect is not supported with system switchover. On system switchover, re-run the bloggerd auto-collect commands on the new Active supervisor to enable auto-collect for respective components.
- Beginning with Cisco NX-OS Release 10.3(3)F, default bloggerd auto-collect is supported for adjmgr, cts, l2fm, and vmtracker.

Default Settings for EEM

This table lists the default settings for EEM parameters.

Parameters	Default
System policies	Active

Configuring EEM

You can create policies that contain actions to take based on system policies. To display information about the system policies, use the **show event manager system-policy** command.

Defining an Environment Variable

You can define a variable to serve as a parameter in an EEM policy.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	event manager environment <i>variable-name</i> <i>variable-value</i> Example: <pre>switch(config)# event manager environment emailto "admin@anyplace.com"</pre>	Creates an environment variable for EEM. The <i>variable-name</i> can be any case-sensitive, alphanumeric string up to 29 characters. The <i>variable-value</i> can be any quoted alphanumeric string up to 39 characters.
Step 3	(Optional) show event manager environment {<i>variable-name</i> all} Example: <pre>switch(config)# show event manager environment all</pre>	Displays information about the configured environment variables.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Defining a User Policy Using the CLI

You can define a user policy using the CLI to the device.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	event manager applet <i>applet-name</i> Example: switch(config)# event manager applet monitorShutdown switch(config-applet)#	Registers the applet with EEM and enters applet configuration mode. The <i>applet-name</i> can be any case-sensitive, alphanumeric string up to 29 characters.
Step 3	(Optional) description <i>policy-description</i> Example: switch(config-applet)# description "Monitors interface shutdown."	Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 4	event <i>event-statement</i> Example: switch(config-applet)# event cli match "conf t ; interface * ; shutdown"	Configures the event statement for the policy. Repeat this step for multiple event statements. See Configuring Event Statements, on page 302 .
Step 5	(Optional) tag <i>tag</i> { and andnot or } <i>tag</i> [and andnot or { <i>tag</i> }] { happens occurs in <i>seconds</i> } Example: switch(config-applet)# tag one or two happens 1 in 10000	Correlates multiple events in the policy. The range for the <i>occurs</i> argument is from 1 to 4294967295. The range for the <i>seconds</i> argument is from 0 to 4294967295 seconds.
Step 6	action <i>number</i> [<i>number2</i>] <i>action-statement</i> Example: switch(config-applet)#action 1.0 cli show interface Ethernet 3/1	Configures an action statement for the policy. Repeat this step for multiple action statements. See Configuring Action Statements, on page 307 .
Step 7	(Optional) show event manager policy-state <i>name</i> [<i>module module-id</i>] Example: switch(config-applet)# show event manager policy-state monitorShutdown	Displays information about the status of the configured policy.
Step 8	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring Event Statements

Use one of the following commands in applet configuration mode to configure an event statement:

Command	Purpose
event application [tag tag] sub-system <i>sub-system-id</i> type <i>event-type</i> Example: <pre>switch(config-applet)# event application sub-system 798 type 1</pre>	Triggers an event when an event specification matches the subsystem ID and application event type. The range for the <i>sub-system-id</i> and for the <i>event-type</i> is from 1 to 4294967295. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. Note To use this command, you must first enable the feature evmed command to enable generic event detectors.
event cli [tag tag] match <i>expression</i> [count <i>repeats</i> time <i>seconds</i>] Example: <pre>switch(config-applet)# event cli match "conf t ; interface * ; shutdown"</pre>	Triggers an event if you enter a command that matches the regular expression. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. The <i>repeats</i> range is from 1 to 65000. The time range, in seconds, is from 0 to 4294967295, where 0 indicates no time limit.
event counter [tag tag] name <i>counter</i> entry-val <i>entry</i> entry-op { eq ge gt le lt ne } [exit-val <i>exit</i> exit-op { eq ge gt le lt ne }] Example: <pre>switch(config-applet)# event counter name mycounter entry-val 20 gt</pre>	Triggers an event if the counter crosses the entry threshold based on the entry operation. The event resets immediately. Optionally, you can configure the event to reset after the counter passes the exit threshold. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. The <i>counter</i> name can be any case-sensitive, alphanumeric string up to 28 characters. The <i>entry</i> and <i>exit</i> value ranges are from 0 to 2147483647.
event fanabsent [fan number] time <i>seconds</i> Example: <pre>switch(config-applet)# event fanabsent time 300</pre>	Triggers an event if a fan is removed from the device for more than the configured time, in seconds. The <i>number</i> range is module-dependent. The <i>seconds</i> range is from 10 to 64000.
event fanbad [fan number] time <i>seconds</i> Example: <pre>switch(config-applet)# event fanbad time 3000</pre>	Triggers an event if a fan fails for more than the configured time, in seconds. The <i>number</i> range is module-dependent. The <i>seconds</i> range is from 10 to 64000.

Command	Purpose
event fib {adjacency extra resource tcam usage route {extra inconsistent missing}} Example: <pre>switch(config-applet)# event fib adjacency extra</pre>	Triggers an event for one of the following: • adjacency extra—If there is an extra route in the unicast FIB. • resource tcam usage—Each time the TCAM utilization percentage becomes a multiple of 5, in either direction. • route {extra inconsistent missing}—If a route is added, changed, or deleted in the unicast FIB.
event gold module {slot all} test test-name [severity {major minor moderate}] testing-type {bootup monitoring ondemand scheduled} consecutive-failure count Example: <pre>switch(config-applet)# event gold module 2 test ASICRegisterCheck testing-type ondemand consecutive-failure 2</pre>	Triggers an event if the named online diagnostic test experiences the configured failure severity for the configured number of consecutive failures. The <i>slot</i> range is from 1 to 10. The <i>test-name</i> is the name of a configured online diagnostic test. The <i>count</i> range is from 1 to 1000.
event memory {critical minor severe} Example: <pre>switch(config-applet)# event memory critical</pre>	Triggers an event if a memory threshold is crossed. See also Configuring Memory Thresholds, on page 311 .
event module [tag tag] status {online offline any} module {all module-num} Example: <pre>switch(config-applet)# event module status offline module all</pre>	Triggers an event if the specified module enters the selected status. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy.
event module-failure [tag tag] type failure-type module {slot all} count repeats [time seconds] Example: <pre>switch(config-applet)# event module-failure type lc-failed module 3 count 1</pre>	Triggers an event if a module experiences the failure type configured. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. The <i>repeats</i> range is from 0 to 4294967295. The <i>seconds</i> range is from 0 to 4294967295, where 0 indicates no time limit.
event none Example: <pre>switch(config-applet)# event none</pre>	Manually runs the policy event without any events specified. Note To use this command, you must first enable the feature evmed command to enable generic event detectors.

Command	Purpose
event oir [tag tag] { fan module powersupply } { anyoir insert remove } [<i>number</i>] Example: <pre>switch(config-applet)# event oir fan remove 4</pre>	Triggers an event if the configured device element (fan, module, or power supply) is inserted or removed from the device. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. You can optionally configure a specific fan, module, or power supply number. The <i>number</i> range is as follows: • Fan number—Module dependent. • Module number—Device dependent. • Power supply number—The range is from 1 to 3.
event policy-default count repeats [<i>time seconds</i>] Example: <pre>switch(config-applet)# event policy-default count 3</pre>	Uses the event configured in the system policy. Use this option for overriding policies. The <i>repeats</i> range is from 1 to 65000. The <i>seconds</i> range is from 0 to 4294967295, where 0 indicates no time limit.
event poweroverbudget Example: <pre>switch(config-applet)# event poweroverbudget</pre>	Triggers an event if the power budget exceeds the capacity of the configured power supplies.
event snmp [tag tag] oid oid get-type { exact next } entry-op { eq ge gt le lt ne } entry-val entry [exit-comb { and or }] exit-op { eq ge gt le lt ne } exit-val exit exit-time time polling-interval interval Example: <pre>switch(config-applet)# event snmp oid 1.3.6.1.2.1.31.1.1.1.6 get-type next entry-op lt 300 entry-val 0 exit-op eq 400 exit-time 30 polling-interval 300</pre>	Triggers an event if the SNMP OID crosses the entry threshold based on the entry operation. The event resets immediately, or optionally you can configure the event to reset after the counter passes the exit threshold. The OID is in dotted decimal notation. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. The <i>entry</i> and <i>exit</i> value ranges are from 0 to 18446744073709551615. The time, in seconds, is from 0 to 2147483647. The interval, in seconds, is from 1 to 2147483647.
event storm-control Example: <pre>switch(config-applet)# event storm-control</pre>	Triggers an event if traffic on a port exceeds the configured storm control threshold.

Command	Purpose
event syslog [<i>occurs count</i>] { <i>pattern string</i> period <i>time</i> priority level tag tag } Example: <pre>switch(config-applet)# event syslog period 500</pre>	Triggers an event if the specified syslog threshold is exceeded. The range for the count is from 1 to 65000, and the range for the time is from 1 to 4294967295. The priority range is from 0 to 7. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy.
event sysmgr memory [<i>module module-num</i>] major <i>major-percent</i> minor <i>minor-percent</i> clear <i>clear-percent</i> Example: <pre>switch(config-applet)# event sysmgr memory minor 80</pre>	Triggers an event if the specified system manager memory threshold is exceeded. The range for the percentage is from 1 to 99.
event sysmgr switchover count <i>count</i> time <i>interval</i> Example: <pre>switch(config-applet)# event sysmgr switchover count 10 time 1000</pre>	Triggers an event if the specified switchover count is exceeded within the time interval specified. The switchover count is from 1 to 65000. The time interval is from 0 to 2147483647.
event temperature [<i>module slot</i>] [<i>sensor-number</i>] threshold { <i>any</i> major minor } Example: <pre>switch(config-applet)# event temperature module 2 threshold any</pre>	Triggers an event if the temperature sensor exceeds the configured threshold. The sensor range is from 1 to 18.

Command	Purpose
event timer {absolute time <i>time</i> name <i>name</i> countdown time <i>time</i> name <i>name</i> cron cronentry <i>string</i> tag <i>tag</i> watchdog time <i>time</i> name <i>name</i>} Example: <pre>switch(config-applet)# event timer absolute time 100 name abtimer</pre>	Triggers an event if the specified time is reached. The range for the time is from 1 to 4294967295. • absolute time—Triggers an event when the specified absolute time of day occurs. • countdown time—Triggers an event when when the specified time counts down to zero. The timer does not reset. • cron cronentry—Triggers an event when the CRON string specification matches the current time. • watchdog time—Triggers an event when the specified time counts down to zero. The timer automatically resets to the initial value and continues to count down. The tag <i>tag</i> keyword-argument pair identifies this specific event when multiple events are included in the policy. Note To use this command, you must first enable the feature evmed command to enable generic event detectors.
event track [tag <i>tag</i>] <i>object-number</i> state {any down up} Example: <pre>switch(config-applet)# event track 1 state down</pre>	Triggers an event if the tracked object is in the configured state. The tag <i>tag</i> keyword-argument pair identifies this specific event when multiple events are included in the policy. The <i>object-number</i> range is from 1 to 500.

Configuring Action Statements

Use the following commands in EEM configuration mode to configure action statements:

Command	Purpose
action <i>number</i> [<i>number2</i>] cli <i>command1</i> [<i>command2...</i>] [local] Example: <pre>switch(config-applet)# action 1.0 cli show interface Ethernet 3/1</pre>	Runs the configured CLI commands. You can optionally run the commands on the module where the event occurred. The action label is in the format <i>number1.number2</i>. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9.

Command	Purpose
action <i>number</i> [. <i>number2</i>] counter name <i>counter value val op {dec inc nop set}</i> Example: <pre>switch(config-applet)# action 2.0 counter name mycounter value 20 op inc</pre>	Modifies the counter by the configured value and operation. The action label is in the format <i>number1.number2</i>. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9. The counter name can be any case-sensitive, alphanumeric string up to 28 characters. The <i>val</i> can be an integer from 0 to 2147483647 or a substituted parameter.
action <i>number</i> [. <i>number2</i>] event-default Example: <pre>switch(config-applet)# action 1.0 event-default</pre>	Executes the default action for the associated event. The action label is in the format <i>number1.number2</i>. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9.
action <i>number</i> [. <i>number2</i>] forceshut [module slot xbar xbar-number] reset-reason seconds Example: <pre>switch(config-applet)# action 1.0 forceshut module 2 reset-reason "flapping links"</pre>	Forces a module, crossbar, or the entire system to shut down. The action label is in the format <i>number1.number2</i>. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9. The reset reason is a quoted alphanumeric string up to 80 characters.
action <i>number</i> [. <i>number2</i>] overbudgetshut [module slot[-slot]] Example: <pre>switch(config-applet)# action 1.0 overbudgetshut module 3-5</pre>	Forces one or more modules or the entire system to shut down because of a power overbudget issue. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9.
action <i>number</i> [. <i>number2</i>] policy-default Example: <pre>switch(config-applet)# action 1.0 policy-default</pre>	Executes the default action for the policy that you are overriding. The action label is in the format <i>number1.number2</i>. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9.
action <i>number</i> [. <i>number2</i>] publish-event Example: <pre>switch(config-applet)# action 1.0 publish-event</pre>	Forces the publication of an application-specific event. The action label is in the format <i>number1.number2</i>. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9.
action <i>number</i> [. <i>number2</i>] reload [module slot[-slot]] Example: <pre>switch(config-applet)# action 1.0 reload module 3-5</pre>	Forces one or more modules or the entire system to reload. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9.

Command	Purpose
action <i>number</i> [<i>.number2</i>] snmp-trap {[<i>intdata1 data</i> [<i>intdata2 data</i>]] [<i>strdata string</i>]} Example: <pre>switch(config-applet)# action 1.0 snmp-trap strdata "temperature problem"</pre>	Sends an SNMP trap with the configured data. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9. The <i>data</i> arguments can be any number up to 80 digits. The <i>string</i> can be any alphanumeric string up to 80 characters.
action <i>number</i> [<i>.number2</i>] syslog [<i>priority prio-val</i>] msg <i>error-message</i> Example: <pre>switch(config-applet)# action 1.0 syslog priority notifications msg "cpu high"</pre>	Sends a customized syslog message at the configured priority. <i>number</i> can be any number up to 16 digits. The range for <i>number2</i> is from 0 to 9. The <i>error-message</i> can be any quoted alphanumeric string up to 80 characters.

**Note**

If you want to allow the triggered event to process any default actions, you must configure the EEM policy to allow the default action. For example, if you match a CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM will not allow the CLI command to execute. You can use the **terminal event-manager bypass** command to allow all EEM policies with CLI matches to execute the CLI command.

Defining a Policy Using a VSH Script

You can define a policy using a VSH script.

Before you begin

Ensure that you are logged in with administrator privileges.

Ensure that your script name is the same name as the script filename.

Procedure

-
- Step 1** In a text editor, list the commands that define the policy.
 - Step 2** Name the text file and save it.
 - Step 3** Copy the file to the following system directory: bootflash://eem/user_script_policies.
-

Registering and Activating a VSH Script Policy

You can register and activate a policy defined in a VSH script.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	event manager policy <i>policy-script</i> Example: switch(config)# event manager policy moduleScript	Registers and activates an EEM script policy. The <i>policy-script</i> can be any case-sensitive alphanumeric string up to 29 characters.
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Overriding a Policy

You can override a system policy.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	(Optional) show event manager policy-state <i>system-policy</i> Example: switch(config-applet)# show event manager policy-state __ethpm_link_flap Policy __ethpm_link_flap Cfg count : 5 Cfg time interval : 10.000000 (seconds) Hash default, Count 0	Displays information about the system policy that you want to override, including thresholds. Use the show event manager system-policy command to find the system policy names. For information about system policies, see Embedded Event Manager System Events and Configuration Examples, on page 583 .
Step 3	event manager applet <i>applet-name</i> override <i>system-policy</i> Example: switch(config)# event manager applet ethport override __ethpm_link_flap switch(config-applet)#	Overrides a system policy and enters applet configuration mode. The <i>applet-name</i> can be any case-sensitive alphanumeric string up to 29 characters. The <i>system-policy</i> must be one of the existing system policies.

	Command or Action	Purpose
Step 4	(Optional) description <i>policy-description</i> Example: description "Overrides link flap policy."	Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 5	Required: [no] event <i>event-statement</i> Example: switch(config-applet)# event policy-default count 2 time 1000	Configures the event statement for the policy. The no form of this command removes the configuration.
Step 6	Required: action <i>number action-statement</i> Example: switch(config-applet)# action 1.0 syslog priority warnings msg "Link is flapping."	Configures an action statement for the policy. Repeat this step for multiple action statements.
Step 7	(Optional) show event manager policy-state <i>name</i> Example: switch(config-applet)# show event manager policy-state ethport	Displays information about the configured policy.
Step 8	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring Memory Thresholds

You can set the memory thresholds that are used to trigger events and set whether the operating system should kill processes if it cannot allocate memory.

Before you begin

Ensure that you are logged in with administrator privileges.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.

	Command or Action	Purpose
Step 2	system memory-thresholds <i>minor minor</i> <i>severe severe critical critical</i> Example: <pre>switch(config)# system memory-thresholds minor 60 severe 70 critical 80</pre>	Configures the system memory thresholds that generate EEM memory events. The default values are as follows: • Minor-85 • Severe-90 • Critical-95 When these memory thresholds are exceeded, the system generates the following syslog: • 2013 May 7 17:06:30 switch %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR • 2013 May 7 17:06:30 switch %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE • 2013 May 7 17:06:30 switch %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL • 2013 May 7 17:06:35 switch %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR ALERT RECOVERED • 2013 May 7 17:06:35 switch %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE ALERT RECOVERED • 2013 May 7 17:06:35 switch %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL ALERT RECOVERED
Step 3	(Optional) system memory-thresholds threshold critical no-process-kill Example: <pre>switch(config)# system memory-thresholds threshold critical no-process-kill</pre>	Configures the system to not kill processes when the memory cannot be allocated. The default value is to allow the system to kill processes, starting with the one that consumes the most memory.
Step 4	(Optional) show running-config include "system memory" Example: <pre>switch(config-applet)# show running-config include "system memory"</pre>	Displays information about the system memory configuration.

	Command or Action	Purpose
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring Syslog as EEM Publisher

You can monitor syslog messages from the switch.



Note The maximum number of searchable strings to monitor syslog messages is 10.

Before you begin

EEM should be available for registration by syslog.

The syslog daemon must be configured and executed.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	event manager applet <i>applet-name</i> Example: <pre>switch(config)# event manager applet abc switch(config-applet)#</pre>	Registers an applet with EEM and enters applet configuration mode.
Step 3	event syslog [<i>tag tag</i>] {<i>occurs number</i> <i>period seconds</i> <i>pattern msg-text</i> <i>priority priority</i>} Example: <pre>switch(config-applet)# event syslog occurs 10</pre>	Monitors syslog messages and invokes the policy based on the search string in the policy. • The tag <i>tag</i> keyword-argument pair identifies this specific event when multiple events are included in the policy. • The occurs <i>number</i> keyword-argument pair specifies the number of occurrences. The range is from 1 to 65000. • The period <i>seconds</i> keyword-argument pair specifies the interval during which the event occurs. The range is from 1 to 4294967295.

	Command or Action	Purpose
		- The pattern <i>msg-text</i> keyword-argument pair specifies the matching regular expression. The pattern can contain character text, an environment variable, or a combination of the two. If the string contains embedded blanks, it is enclosed in quotation marks. - The priority <i>priority</i> keyword-argument pair specifies the priority of the syslog messages. If this keyword is not selected, all syslog messages are set at the informational priority level.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Verifying the EEM Configuration

To display EEM configuration information, perform one of the following tasks:

Command	Purpose
show event manager environment [<i>variable-name</i> all]	Displays information about the event manager environment variables.
show event manager event-types [<i>event</i> all module slot]	Displays information about the event manager event types.
show event manager history events [detail] [maximum <i>num-events</i>] [severity { catastrophic minor moderate severe }]	Displays the history of events for all policies.
show event manager policy-state <i>policy-name</i>	Displays information about the policy state, including thresholds.
show event manager script system [<i>policy-name</i> all]	Displays information about the script policies.
show event manager system-policy [all]	Displays information about the predefined system policies.
show running-config eem	Displays information about the running configuration for EEM.
show startup-config eem	Displays information about the startup configuration for EEM.

Configuration Examples for EEM

This example shows how to override the `__lcm_module_failure` system policy by changing the threshold for just module 3 hitless upgrade failures. This example also sends a syslog message. The settings in the system policy, `__lcm_module_failure`, apply in all other cases.

```
event manager applet example2 override __lcm_module_failure
event module-failure type hitless-upgrade-failure module 3 count 2
action 1 syslog priority errors msg module 3 "upgrade is not a hitless upgrade!"
action 2 policy-default
```

This example shows how to override the `__ethpm_link_flap` system policy and shuts down the interface:

```
event manager applet ethport override __ethpm_link_flap
event policy-default count 2 time 1000
action 1 cli conf t
action 2 cli int et1/1
action 3 cli no shut
```

This example creates an EEM policy that allows the CLI command to execute but triggers an SNMP notification when a user enters configuration mode on the device:

```
event manager applet TEST
event cli match "conf t"
action 1.0 snmp-trap strdata "Configuration change"
action 2.0 event-default
```



Note You must add the **event-default** action statement to the EEM policy or EEM will not allow the CLI command to execute.

This example shows how to correlate multiple events in an EEM policy and execute the policy based on a combination of the event triggers. In this example, the EEM policy is triggered if one of the specified syslog patterns occurs within 120 seconds.

```
event manager applet eem-correlate
event syslog tag one pattern "copy bootflash:.* running-config.*"
event syslog tag two pattern "copy run start"
event syslog tag three pattern "hello"
tag one or two or three happens 1 in 120
action 1.0 reload module 1
```

Upon reaching a maximum failure threshold, the AsicMemory, FpgaRegTest, and L2ACLRedirect system policies force a reload of the switch. This example shows how to override the default action for one of these policies and issue a syslog instead:

```
event manager applet gold override __fpgareg
action 1 syslog priority emergencies msg FpgaRegTest_override
```

This example shows how to override a default policy but still enact the default action:

```
event manager applet gold_fpga_ovrd override __fpgareg
  action 1 policy-default
  action 2 syslog priority emergencies msg FpgaRegTest_override
```



Note For additional EEM configuration examples, see [Embedded Event Manager System Events and Configuration Examples, on page 583](#).

Event Log Auto-Collection and Backup

Automatically collected event logs are stored locally on switch memory. Event log file storage is a temporary buffer that stores files for a fixed amount of time. Once the time period has elapsed, a roll-over of the buffer makes room for the next files. The roll-over uses a first-in-first-out method.

Beginning with Cisco NX-OS Release 9.3(3), EEM uses the following methods of collection and backup:

- Extended Log File Retention
- Trigger-Based Event Log Auto-Collection

Extended Log File Retention

Beginning with Cisco NX-OS release 9.3(3), all Cisco Nexus platform switches, with at least 8Gb of system memory, support the extended retention of event logging files. Storing the log files locally on the switch or remotely through an external container, reduces the loss of event logs due to rollover.

Enabling Extended Log File Retention For All Services

Extended Log File Retention is enabled by default for all services running on a switch. If the switch doesn't have the log file retention feature enabled (**no bloggerd log-dump** is configured), use the following procedure to enable it.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	bloggerd log-dump all Example: <pre>switch(config)# bloggerd log-dump all switch(config)#</pre>	Enables the log file retention feature for all services.

Example

```
switch# configure terminal
switch(config)# bloggerd log-dump all
Sending Enable Request to Bloggerd
```

```
Bloggerd Log Dump Successfully enabled
switch(config)#
```

Disabling Extended Log File Retention For All Services

Extended Log File Retention is enabled by default for all services running on the switch.

If the switch has the log file retention feature enabled for all services, and you want to disable it, use the following procedure.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	no bloggerd log-dump all Example: <pre>switch(config)# no bloggerd log-dump all switch(config)#</pre>	Disables the log file retention feature for all services on the switch.

Example

```
switch# configure terminal
switch(config)# no bloggerd log-dump all
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)#
```

Enabling Extended Log File Retention For a Single Service

Use this procedure to enable Log File Retention for a single service.

Procedure

	Command or Action	Purpose
Step 1	show system internal sysmgr service name <i>service-type</i> Example: <pre>switch# show system internal sysmgr service name aclmgr</pre>	Displays information about the ACL Manager including the service SAP number.
Step 2	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	bloggerd log-dump sap <i>number</i> Example: <pre>switch(config)# bloggerd log-dump sap 351</pre>	Enables the log file retention feature for the ACL Manager service.
Step 4	show system internal bloggerd info log-dump-info Example: <pre>switch(config)# show system internal bloggerd info log-dump-info</pre>	Displays information about the log file retention feature on the switch.

Example

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
UUID = 0x182, PID = 653, SAP = 351
State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
Restart count: 1
Time of last restart: Mon Nov  4 11:10:39 2019.
The service never crashed since the last reboot.
Tag = N/A
Plugin ID: 0
switch(config)# configure terminal
switch(config)# bloggerd log-dump sap 351
Sending Enable Request to Bloggerd
Bloggerd Log Dump Successfully enabled
switch(config)# show system internal bloggerd info log-dump-info
-----
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
-----
Module      | VDC      | SAP      | Enabled?
-----
1           | 1        | 351 (MTS_SAP_ACLMGR) | Enabled
-----
Log Dump Throttle Switch-Wide Config:
-----
Log Dump Throttle                               : ENABLED
Minimum buffer rollover count (before throttling) : 5
Maximum allowed rollover count per minute        : 1
-----

switch(config)#
```

Displaying Extended Log Files

Use this task to display the event log files currently stored on the switch.

Procedure

	Command or Action	Purpose
Step 1	dir debug:log-dump/ Example: switch# dir debug:log-dump/	Displays the event log files currently stored on the switch.

Example

```
switch# dir debug:log-dump/

3676160 Dec 05 02:43:01 2019 20191205023755_evtlog_archive.tar
3553280 Dec 05 06:05:06 2019 20191205060005_evtlog_archive.tar

Usage for debug://sup-local
913408 bytes used
4329472 bytes free
5242880 bytes total
```

Displaying Global Dictionary Per Log Statistics

This CLI displays the statistics of log message being logged by each component with a counter, to store the number of times a log being repeated from the system up time.

Procedure

	Command or Action	Purpose
Step 1	show system internal sdwrap buffers sap <sap-num> dict-stats detailed Example: switch# show system internal sdwrap buffers sap <sap-num> dict-stats detailed	Displays the per log statistics of each component.

Example

```
switch# show system internal sdwrap buffers sap 221 dict-stats detailed

Sap received is: 221

SDWrap Format Strings Dictionary stats for sap MTS_SAP_L2FM (221)

UUID: SRVUUID_LIBSDWRAP, Inst Type: 0

MsgId Frequency Message
-----
 4      1 System is not undergoing ISSU
78      1 Vlan %d is part of reserved vlan bmp from sdb          179      1 Vlan
%d is not found in L2FM database. Skipping the delete request 306      1 Vlan %d is removed
```

```

from L2FM database and MTM database
416 1 mts_drap_get_my_local_swid_only_msg failed with rc %x
496 1 Lookup for backplane mac failed for vdc %d with st = %s
598 1 L2FM - Slot %d SwCardId %d Port %d - %d Fp %d Cli %d

```

Disabling Extended Log File Retention For a Single Service

Extended Log File Retention is enabled by default for all services on the switch from release 9.3(5). If the switch has the log file retention feature enabled for a single service or all services, and you want to disable a specific service or services, use the following procedure.

Procedure

	Command or Action	Purpose
Step 1	show system internal sysmgr service name <i>service-type</i> Example: <pre>switch# show system internal sysmgr service name aclmgr</pre>	Displays information about the ACL Manager including the service SAP number.
Step 2	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 3	no bloggerd log-dump sap number Example: <pre>switch(config)# no bloggerd log-dump sap 351</pre>	Disables the log file retention feature for the ACL Manager service.
Step 4	show system internal bloggerd info log-dump-info Example: <pre>switch(config)# show system internal bloggerd info log-dump-info</pre>	Displays information about the log file retention feature on the switch.

Example

The following example shows how to disable extended log file retention for a service named "aclmgr":

```

switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
  UUID = 0x182, PID = 653, SAP = 351
  State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
  Restart count: 1
  Time of last restart: Mon Nov  4 11:10:39 2019.
  The service never crashed since the last reboot.
  Tag = N/A
  Plugin ID: 0
switch(config)# configure terminal

```

```

switch(config)# no bloggerd log-dump sap 351
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)# show system internal bloggerd info log-dump-info
-----
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
-----
Module      | VDC      | SAP              | Enabled?
-----
1           | 1        | 351 (MTS_SAP_ACLMGR) | Disabled
-----

Log Dump Throttle Switch-Wide Config:
-----
Log Dump Throttle                               : ENABLED
Minimum buffer rollover count (before throttling) : 5
Maximum allowed rollover count per minute         : 1
-----

switch(config)#

```

Trigger-Based Event Log Auto-Collection

Trigger-based log collection capabilities:

- Automatically collect relevant data when issues occur.
- No impact on control plane
- Customizable configuration:
 - Defaults populated by Cisco
 - Selectively override what-to-collect by network administrator or by Cisco TAC.
 - Automatically update new triggers on image upgrades.
- Store logs locally on the switch or remotely on an external server.
- Supports severity 0, 1, and 2 syslogs
- Custom syslogs for ad-hoc events (auto-collection commands attached to the syslogs)

Enabling Trigger-Based Log File Auto-Collection

To enable trigger-based automatic creation of log files, you must create an override policy for the `__syslog_trigger_default` system policy with a custom YAML file and define the specific logs for which information will be collected.

For more information on creating a custom YAML file to enable log file auto-collection, see [Configuring the Auto-Collection YAML File, on page 325](#).

Log-Profile YAML File

The Log-Profile YAML file is used to define the throttle limit for any component. The `log_profile.yaml` file is located in the switch directory: `/bootflash`.

The Bloggerd maintains component name and rollovers information and stores/retains the log files based on the limits that are defined in the global YAML file for specific components.

By default the switch comes with a throttle value of 5. You can add an entry in the **log_profile.yaml** file to override the throttle count.

To reflect the changes made in `/bootflash/log_profile.yaml` file, execute the following CLI during run time at bloggerd:

- switch# bloggerd reparse log-profile

Example Log-Profile YAML File

The following is an example of a default `log_profile.yaml` file which is packaged part of the image. The definitions for the keys/values in the file are in the table that follows.

```
273:
  entry_1:
    srv_uuid: 273
    instance: 0
    rollovers_allowed: 250
    rotations_allowed: 5
    mod: sup

274:
  entry_1:
    srv_uuid: 274
    instance: 0
    rollovers_allowed: 250
    rotations_allowed: 5
    mod: sup
```

Key: Value	Description
273	UUID of the component whose sdwrap buffer throttling needs to be overridden.
entry_1:	Only one entry supported per components Upto 20 entries can be made per component. Each entry is identified entry_1 through entry_20 .
srv_uuid:	Each sdwrap log buffer is identified with (uuid, instance id) tuple.
instance:	Sdwrap log buffer instance id wrt srv_uuid field above. A "-1" means, all instances.
rollovers_allowed:	How many rollovers allowed per minute. 0-500 allowed value.
rotations_allowed:	How many rotations allowed per throttle.
mod:	Name of the syslog component (<code>platform</code> is a facility name in syslog).

Auto-Collection YAML File

The Auto-Collection YAML file that is specified in the **action** command in the EEM function, defines actions for different system or feature components. This file is located in the switch directory: `/bootflash/scripts`. In addition to the default YAML file, you can create component-specific YAML files and place them in the same directory. The naming convention for component-specific YAML files is **component-name.yaml**. If a component-specific file is present in the same directory, it takes precedence over the file that is specified in

the **action** command. For example, if the action file, **bootflash/scripts/platform.yaml** is in the **/bootflash/scripts** directory with the default action file, **bootflash/scripts/test.yaml**, then the instructions defined in **platform.yaml** file take precedence over the instructions for the platform component present in the default **test.yaml** file.

Examples of components are, ARP, BGP, IS-IS, and so on. If you are not familiar with all the component names, contact Cisco Customer Support for assistance in defining the YAML file for component-specific actions (and for the default **test.yaml** file as well).

Example:

```
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml $_syslog_msg
```

Creating or Deleting Auto-Collection Per Component

Beginning with Cisco NX-OS Release 10.2(2)F, the auto-collect adoption improvement feature allows you to control the auto-collection for a single or set of components based on your requirement. You can use the following command for creation or deletion of auto-collect YAML files.



Note Beginning with Cisco NX-OS Release 10.3(1)F, multiple components are enabled by default and the YAML file of the component is copied to the default auto-collect folder. However, you can disable and enable bloggerd auto-collect component using this command.



Note The YAML file is editatble and handle it with caution. If the file gets corrupted with any syntax, tar will not be generated. .

```
switch# bloggerd auto-collect component <component_name> {enable | disable}
```

When you use the enable command, the YAML file of the component is copied from the backup folder to the default auto-collect folder. Note that you cannot copy the contents of the backup-staging folder as it is a read-only folder; whereas, you can copy the contents of the default auto-collect folder (**bootflash:scripts** folder), if required.

When you use the disable command, the YAML file of the component is removed from the default auto-collect folder under the **bootflash:scripts** folder.

A sample output is as follows:

```
switch# bloggerd auto-collect component arp enable
Component arp auto-collect successfully enabled.
arp.yaml file copied from /bootflash/scripts/backup-staging to
/bootflash/scripts/default-autocollect
switch# dir bootflash:scripts/default-autocollect
435 Nov 10 08:43:21 2021 arp.yaml
438 Oct 25 05:55:11 2021 fex.yaml
579 Oct 25 05:55:11 2021 kern.yaml
Usage for bootflash://sup-local
11078049792 bytes used
10653151232 bytes free
21731201024 bytes total
switch# dir bootflash:scripts/backup-staging/
switch# bloggerd auto-collect component ?
  CrdCfg          Auto-collect for CRDCFG
  aclmgr          Auto-collect for ACLMgr
```

Creating or Deleting Auto-Collection Per Component

aclqos	Auto-collect for ACLQOS
adjmgr	Auto-collect for Adjacency Manager
arp	Auto-collect for ARP
bcm_usd	Auto-collect for BCM USD
bgp	Auto-collect for BGP
cardclient	Auto-collect for CARD CLIENT
cdp	Auto-collect for CPD
cfs	Auto-collect for CFS
clis	Auto-collect for CLIS
cts	Auto-collect for CTS
dhcp_snoop	Auto-collect for DHCP Snoop
eigrp	Auto-collect for EIGRP
eltn	Auto-collect for ELTM
ethport	Auto-collect for Eth Port Manager
feature-mgr	Auto-collect for Feature Manager
fex	Auto-collect for Fex (Satellite Manager)
hmm	Auto-collect for HMM
hsrp_engine	Auto-collect for HSRP
icam	Auto-collect for ICAM
icmpv6	Auto-collect for ICMPv6
iftmc	Auto-collect for IFTMC
im	Auto-collect for IM
ip	Auto-collect for IP
ipfib	Auto-collect for IPFIB Manager
isis	Auto-collect for ISIS
jer_usd	Auto-collect for JER USD
kafka	Auto-collect for KAFKA Manager
kern	Auto-collect for Kernel
l2fm	Auto-collect for L2FM
l2rib	Auto-collect for L2RIB
l3vm	Auto-collect for L3VM
lacp	Auto-collect for LACP
lldp	Auto-collect for LLDP
m2rib	Auto-collect for M2RIB
mfdm	Auto-collect for MFDM
mrib	Auto-collect for MRIB
nbm	Auto-collect for NBM Daemon
netstack	Auto-collect for Netstack
ngoam	Auto-collect for NGOAM
nve	Auto-collect for NVE
ospf	Auto-collect for Open Shortest Path First Unicast Routing Protocol (OSPF)
ospfv3	Auto-collect for Open Shortest Path First Version 3 Unicast Routing Protocol
pfma	Auto-collect for PFM
pim	Auto-collect for PIM
pktmgr	Auto-collect for Packet Manager
pltfm_config	Auto-collect for PLTFM CONFIG
port-channel	Auto-collect for Port Channel Manager
qos	Auto-collect for QOS Manager
rip	Auto-collect for RIP
sdaa	Auto-collect for SDAA
sla_responder	Auto-collect for SLA Responder
sla_sender	Auto-collect for SLA Sender
sla_twamp	Auto-collect for SLA Twamp
smm	Auto-collect for SMM
snmpmib_proc	Auto-collect for Snmpmib_proc
spm	Auto-collect for SPM
statsclient	Auto-collect for Statistics Client
sysmgr	Auto-collect for SYSMGR
tahusd	Auto-collect for TAHUSD
tctrl_usd	Auto-collect for TCTRL USD
tun_enc_mgr	Auto-collect for TEM
udld	Auto-collect for UDLD
ufdm	Auto-collect for UFDM

```

vmtracker      Auto-collect for VMTRACKER
vntag_mgr      Auto-collect for VNTAG Mgr
vpc            Auto-collect for VPC
vrrp-cfg       Auto-collect for VRRP Configuration
vrrp-eng       Auto-collect for VRRP Engine
vrrpv3         Auto-collect for VRRPV3
Usage for bootflash://sup-local 11078049792 bytes used
10653151232 bytes free
21731201024 bytes total
switch# dir bootflash:scripts/default-autocollect^C n9k-A# dir
bootflash:scripts/default-autocollect
435 Nov 10 08:43:21 2021 arp.yaml
438 Oct 25 05:55:11 2021 fex.yaml
579 Oct 25 05:55:11 2021 kern.yaml Usage for bootflash://sup-local 11078049792 bytes used
10653151232 bytes free
21731201024 bytes total

```

The following is an example to create pre-populated YAML file for the UDLD component.

```

n9k-A# bloggerd auto-collect component udld enable
Component udld auto-collect successfully enabled.
udld.yaml file copied from /bootflash/scripts/backup-staging to
/bootflash/scripts/default-autocollect
n9k-A# dir bootflash:scripts/default-autocollect
435 Nov 10 08:43:21 2021 arp.yaml
438 Oct 25 05:55:11 2021 fex.yaml
579 Oct 25 05:55:11 2021 kern.yaml
431 Nov 10 08:44:45 2021 udld.yaml
Usage for bootflash://sup-local
11078053888 bytes used
10653147136 bytes free
21731201024 bytes total
n9k-A# sh running-config all | include bloggerd
bloggerd log-dump all
bloggerd log-throttle
no bloggerd log-transfer

```

The following is an example to delete pre-populated YAML file for the UDLD component.

```

n9k-A# bloggerd auto-collect component udld disable
Component udld auto-collect successfully disabled.
udld.yaml file deleted from /bootflash/scripts/default-autocollect
n9k-A# dir bootflash:scripts/default-autocollect
435 Nov 10 08:43:21 2021 arp.yaml
438 Oct 25 05:55:11 2021 fex.yaml
579 Oct 25 05:55:11 2021 kern.yaml
Usage for bootflash://sup-local
11078049792 bytes used
10653151232 bytes free
21731201024 bytes total
n9k-A#

```

Configuring the Auto-Collection YAML File

The contents of a YAML file determines the data collected during trigger-based auto-collection. There must be only one YAML file on the switch but it can contain auto-collection meta-data for any number of switch components and messages.

Locate the YAML file in the following directory on the switch:

```
/bootflash/scripts
```

Invoke the YAML file for trigger-based collection by using the following example. The example shows the minimum required configuration for trigger-based collection to work with a user-defined YAML file.

```
switch# show running-config eem
!Command: show running-config eem
!Running configuration last done at: Mon Sep 30 19:34:54 2019
!Time: Mon Sep 30 22:24:55 2019
version 9.3(3) Bios:version 07.59
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml $_syslog_msg
```

In the preceding example, "test_1" is the name of the applet and "test.yaml" is the name of the user-configured YAML file present in the /bootflash/scripts directory.

Example YAML File

The following is an example of a basic YAML file supporting the trigger-based event log auto-collection feature. The definitions for the keys/values in the file are in the table that follows.



Note Make sure that the YAML file has proper indentation. As a best practice, run it through any "online YAML validator" before using it on a switch.

```
bash-4.3$ cat /bootflash/scripts/test.yaml
version: 1
components:
  securityd:
    default:
      tech-sup: port
      commands: show module
  platform:
    default:
      tech-sup: port
      commands: show module
```

Key: Value	Description
version: 1	Set to 1. Any other number creates an incompatibility for the auto collect script.
components:	Keyword specifying that what follows are switch components.
securityd:	Name of the syslog component (<i>securityd</i> is a facility name in syslog).
default:	Identifies all messages belonging to the component.
tech-sup: port	Collect tech support of the port module for the <i>securityd</i> syslog component.
commands: show module	Collect show module command output for the <i>securityd</i> syslog component.
platform:	Name of the syslog component (<i>platform</i> is a facility name in syslog).
tech-sup: port	Collect tech support of the port module for the <i>platform</i> syslog component.
commands: show module	Collect show module command output for the <i>platform</i> syslog component.

Use the following example to associate auto-collect metadata only for a specific log. For example, SECURITYD-2-FEATURE_ENABLE_DISABLE

```
securityd:
    feature_enable_disable:
        tech-sup: security
        commands: show module
```

Key: Value	Description
securityd:	Name of the syslog component (<code>securityd</code> is a facility name in syslog).
feature_enable_disable:	Message ID of the syslog message.
tech-sup: security	Collect tech support of the security module for the <code>securityd</code> syslog component.
commands: show module	Collect show module command output for the security syslog component.

Example syslog output for the above YAML entry:

```
2019 Dec 4 12:41:01 n9k-c93108tc-fx %SECURITYD-2-FEATURE_ENABLE_DISABLE: User
has enabled the feature bash-shell
```

Use the following example to specify multiple values.

```
version: 1
components:
    securityd:
        default:
            commands: show module;show version;show module
            tech-sup: port;lldp
```



Note Use semicolons to separate multiple show commands and tech support key values (see the preceding example).

Beginning with Release 10.1(1), `test.yaml` can be replaced with a folder inside which more than one YAML files can be present. All the YAML files in the folder must follow the `ComponentName.yaml` naming convention.

In the following example, `test.yaml` is replaced with `test_folder`:

```
test.yaml:
event manager applet logging2 override __syslog_trigger_default
    action 1.0 collect test.yaml rate-limit 30 $_syslog_msg

test_folder:
event manager applet logging2 override __syslog_trigger_default
    action 1.0 collect test_folder rate-limit 30 $_syslog_msg
```

The following example shows the path and component(s) for `test_folder`:

```
ls /bootflash/scripts/test_folder
bgp.yaml ppm.yaml
```

Limiting the Amount of Auto-Collections Per Component

For auto-collection, the limit of the number of bundles per component event is set to one (1) by default from Cisco NX-OS Release 10.2(2)F. Earlier, this limit was three (3) by default. If more than the default events occur for a component, then the events are dropped with the status message `EVENTLOGLIMITREACHED`. The auto-collection of the component event restarts when the event log has rolled over.

Example:

```
switch# show system internal event-logs auto-collect history
DateTime                Snapshot ID  Syslog                Status/Secs/Logsize(Bytes)
2020-Jun-27 07:20:03    1140276903  ACLMGR-0-TEST_SYSLOG  EVENTLOGLIMITREACHED
2020-Jun-27 07:15:14    1026359228  ACLMGR-0-TEST_SYSLOG  RATELIMITED
2020-Jun-27 07:15:09    384952880   ACLMGR-0-TEST_SYSLOG  RATELIMITED
2020-Jun-27 07:13:55    1679333688  ACLMGR-0-TEST_SYSLOG  PROCESSED:2:9332278
2020-Jun-27 07:13:52    1679333688  ACLMGR-0-TEST_SYSLOG  PROCESSING
2020-Jun-27 07:12:55    502545693   ACLMGR-0-TEST_SYSLOG  RATELIMITED
2020-Jun-27 07:12:25    1718497217  ACLMGR-0-TEST_SYSLOG  RATELIMITED
2020-Jun-27 07:08:25    1432687513  ACLMGR-0-TEST_SYSLOG  PROCESSED:2:10453823
2020-Jun-27 07:08:22    1432687513  ACLMGR-0-TEST_SYSLOG  PROCESSING
2020-Jun-27 07:06:16    90042807    ACLMGR-0-TEST_SYSLOG  RATELIMITED
2020-Jun-27 07:03:26    1737578642  ACLMGR-0-TEST_SYSLOG  RATELIMITED
2020-Jun-27 07:02:56    40101277    ACLMGR-0-TEST_SYSLOG  PROCESSED:3:10542045
2020-Jun-27 07:02:52    40101277    ACLMGR-0-TEST_SYSLOG  PROCESSING
```

Auto-Collection Log Files

About Auto-Collection Log Files

The configuration in a YAML file determines the contents of an auto-collected log file. You can't configure the amount of memory used for collected log files. You can configure the frequency of when the stored files get purged.

Autocollected log files get saved in the following directory:

```
switch# dir bootflash:eem_snapshots
 44205843      Sep 25 11:08:04 2019
1480625546_SECURITYD_2_FEATURE_ENABLE_DISABLE_eem_snapshot.tar.gz
  Usage for bootflash://sup-local
 6940545024 bytes used
44829761536 bytes free
51770306560 bytes total
```

Accessing the Log Files

Locate the logs by using the command keyword "debug":

```
switch# dir debug:///
...
 26   Oct 22 10:46:31 2019  log-dump
 24   Oct 22 10:46:31 2019  log-snapshot-auto
 26   Oct 22 10:46:31 2019  log-snapshot-user
```

The following table describes the log locations and the log types stored.

Location	Description
log-dump	This folder stores Event logs on log rollover.
log-snapshot-auto	This folder contains the auto-collected logs for syslog events 0, 1, 2.
log-snapshot-user	This folder stores the collected logs when you run the <code>bloggerd log-snapshot <></code> command.

Use the following example to view the log files generated on log rollover:

```
switch# dir debug:log-dump/
debug:log-dump/20191022104656_evtlog_archive.tar
debug:log-dump/20191022111241_evtlog_archive.tar
debug:log-dump/20191022111841_evtlog_archive.tar
debug:log-dump/20191022112431_evtlog_archive.tar
debug:log-dump/20191022113042_evtlog_archive.tar
debug:log-dump/20191022113603_evtlog_archive.tar
```

Parsing the Log tar Files

Use the following example to parse the logs in the tar files:

```
switch# show system internal event-logs parse debug:log-dump/20191022104656_evtlog_archive.tar
-----LOGS:/tmp/BLOGGERD0.991453012199/tmp/1-191022104658-191022110741-device_test-M27-V1-I1:0-P884.gz-----
2019 Oct 22 11:07:41.597864 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Data Space
Limits(bytes): Soft: -1  Ha rd: -1
2019 Oct 22 11:07:41.597857 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Stack Space
Limits(bytes): Soft: 500000  Hard: 500000
2019 Oct 22 11:07:41.597850 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):AS: 1005952076
-1
2019 Oct 22 11:07:41.597406 E_DEBUG Oct 22 11:07:41 2019(device_test_process_events):Sdwrap
msg unknown
2019 Oct 22 11:07:41.597398 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Going back to
select
2019 Oct 22 11:07:41.597395 E_DEBUG Oct 22 11:07:41 2019(nvram_test):TestNvram examine 27
blocks
2019 Oct 22 11:07:41.597371 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Parent: Thread
created test index:4 thread_id:-707265728
2019 Oct 22 11:07:41.597333 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Node inserted
2019 Oct 22 11:07:41.597328 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):The test index
in diag is 4
2019 Oct 22 11:07:41.597322 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):result severity
level
2019 Oct 22 11:07:41.597316 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):callhome alert
level
```

The following table describes the additional keywords available for parsing the specific tar file:

Keyword	Description
component	Decode logs belonging to the component identified by process name.
from-datetime	Decode logs from a specific date and time in yy[mm[dd[HH[MM[SS]]]]]] format.
instance	List of SDWRAP buffer instances to be decoded (comma separated).
module	Decode logs from modules such as SUP and LC (using module IDs).
to-datetime	Decode logs up to a specific date and time in yy[mm[dd[HH[MM[SS]]]]]] format.

Copying Logs to a Different Location

Use the following example to copy logs to a different location such as a remote server:

```
switch# copy debug:log-dump/20191022104656_evtlog_archive.tar
scp://<ip-address>/nobackup/<user> vrf management use-kstack
Enter username: user@<ip-address>'s password:
20191022104656_evtlog_archive.tar                                100% 130KB
130.0KB/s 00:00
Copy complete, now saving to disk (please wait)...
Copy complete.
```

Purging Auto-Collection Log Files

There are two types of generated trigger-based auto-collection logs: EventHistory and EventBundle.

Purge Logic for EventHistory Logs

For event history, purging occurs in the /var/sysmgr/srv_logs/xport folder. 250MB of partitioned RAM is mounted at /var/sysmgr/srv_logs directory.

If the /var/sysmgr/srv_logs memory usage is under 65% of the 250MB allocated, no files get purged. When the memory utilization reaches the 65% limit level, the oldest files get purged until there's enough memory available to continue saving new logs.

Purge Logic for EventBundle Logs

For event bundles, the purge logic occurs in the /bootflash/eem_snapshots folder. For storing the auto-collected snapshots, the EEM auto-collect script allocates 5% of the bootflash storage. The logs get purged once the 5% bootflash capacity is used.

When a new auto-collected log is available but there's no space to save it in bootflash (already at 5% capacity), the system checks the following:

1. If there are existing auto-collected files that are more than 12 hours old, the system deletes the files and the new logs get copied.
2. If the existing auto collected files are less than 12 hours old, the system discards the newly collected logs without saving them.

You can modify the 12-hour default purge time by using the following commands. The time specified in the command is in minutes.

```
switch(config)# event manager applet test override __syslog_trigger_default
switch(config-applet)# action 1.0 collect test.yaml purge-time 300 $_syslog_msg
```

event manager command: *test* is an example name for the policy. **__syslog_trigger_default** is the name of the system policy that you want to override. This name must begin with a double underscore (__).

action command: **1.0** is an example number for the order in which the action is executed. **collect** indicates that data is collected using the YAML file. *test.yaml* is an example name of the YAML file. **\$_syslog_msg** is the name of the component.



Note At any given time, there can be only one trigger-based auto-collection event in progress. If another new log event is attempting to be stored when auto-collection is already occurring, the new log event is discarded.

By default, there's only one trigger-based bundle collected every five minutes (300 sec). This rate limiting is also configurable by the following commands. The time specified in the command is in seconds.

```
switch(config)# event manager applet test override __syslog_trigger_default
switch(config-applet)# action 1.0 collect test.yaml rate-limit 600 $_syslog_msg
```

event manager command: *test* is an example name for the policy. **__syslog_trigger_default** is an example name of the system policy to override. This name must begin with a double underscore (__).

action command: **1.0** is an example number for the order in which the action is executed. **collect** indicates that data is collected using the YAML file. *test.yaml* is an example name of the YAML file. **\$_syslog_msg** is the name of the component.

Beginning with Release 10.1(1), the rate of collection can also be regulated using a maximum number of triggers option, ensuring that only those many number of triggers are honored. After the **max-triggers** value is reached, no more bundles will be collected on the syslog occurrence.

```
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml rate-limit 30 max-triggers 5 $_syslog_msg
```



Note If you delete auto collected bundles manually from `debug:log-snapshot-auto/`, then it will restart the collection based on the configured number of **max-triggers** when the next event occurs.

Auto-Collection Statistics and History

The following example shows trigger-based collection statistics:

```
switch# show system internal event-logs auto-collect statistics
-----EEM Auto Collection Statistics-----
Syslog Parse Successful :88 Syslog Parse Failure :0
Syslog Ratelimited :0 Rate Limit Check Failed :0
Syslog Dropped(Last Action In Prog) :53 Storage Limit Reached :0
User Yaml Action File Unavailable :0 User Yaml Parse Successful :35
User Yaml Parse Error :0 Sys Yaml Action File Unavailable :11
Sys Yaml Parse Successful :3 Sys Yaml Parse Error :0
Yaml Action Not Defined :0 Syslog Processing Initiated :24
Log Collection Failed :0 Tar Creation Error :0
Signal Interrupt :0 Script Exception :0
Syslog Processed Successfully :24 Logfiles Purged :0
```

The following example shows trigger-based collection history (the processed syslogs, process time, size of the data collected) obtained using a CLI command:

```
switch# show system internal event-logs auto-collect history
DateTime Snapshot ID Syslog Status/Secs/Logsize(Bytes)
2019-Dec-04 05:30:32 1310232084 VPC-0-TEST_SYSLOG PROCESSED:9:22312929
2019-Dec-04 05:30:22 1310232084 VPC-0-TEST_SYSLOG PROCESSING
2019-Dec-04 04:30:13 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSED:173:33194665
2019-Dec-04 04:28:47 897805674 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:28:47 947981421 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:27:19 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSING
2019-Dec-04 02:17:16 1957148102 CARDCLIENT-2-FPGA_BOOT_GOLDEN NOYAMLFILFOUND
```

Verifying Trigger-Based Log Collection

Verify that the trigger-based log collection feature is enabled by entering the **show event manager system-policy | i trigger** command as in this example:

```
switch# show event manager system-policy | i trigger n 2
      Name : __syslog_trigger_default
      Description : Default policy for trigger based logging
      Overridable : Yes
      Event type : 0x2101
```

Checking Trigger-Based Log File Generation

You can check to see if the trigger-based auto-collection feature has generated any event log files. Enter one of the commands in the following examples:

```
switch# dir bootflash:eem_snapshots
9162547 Nov 12 22:33:15 2019 1006309316_SECURITYD_2_FEATURE_ENABLE_DISABLE_eem_snapshot.tar.gz
```

```

Usage for bootflash://sup-local
8911929344 bytes used
3555950592 bytes free
12467879936 bytes total

switch# dir debug:log-snapshot-auto/
63435992 Dec 03 06:28:52 2019
20191203062841_1394408030_PLATFORM_2_MOD_PWRDN_eem_snapshot.tar.gz

Usage for debug://sup-local
544768 bytes used
4698112 bytes free
5242880 bytes total

```

Local Log File Storage

Local log file storage capabilities:

- Amount of local data storage time depends on the scale, and type, of deployment. For both modular and nonmodular switches, the storage time is from 15 minutes to several hours of data. To be able to collect relevant logs that span a longer period:
 - Only enable event log retention for the specific services/features you need. See [Enabling Extended Log File Retention For a Single Service , on page 317](#).
 - Export the internal event logs off the switch. See [External Log File Storage, on page 334](#).
- Compressed logs are stored in RAM.
- 250MB memory is reserved for log file storage.
- Log files are optimized in tar format (one file for every five minutes or 10MB, whichever occurs first).
- Allow snap-shot collection.

Generating a Local Copy of Recent Log Files

Extended Log File Retention is enabled by default for all services running on a switch. Log files are stored locally on flash memory. Use the following procedure to generate a file of up to ten of the most recent event log files.

Procedure

	Command or Action	Purpose
Step 1	bloggerd log-snapshot [<i>file-name</i>] [bootflash: <i>file-path</i> logflash: <i>file-path</i> usb1:] [size <i>file-size</i>] [time <i>minutes</i>] Example: switch# bloggerd log-snapshot snapshot1	Creates a snapshot bundle file of the last ten event logs stored on the switch. Default storage for this operation is logflash. <i>file-name</i>: The filename of the generated snapshot log file bundle. Use a maximum of 64 characters for <i>file-name</i>. Note This variable is optional. If it is not configured, the system applies a timestamp and

	Command or Action	Purpose
		"_snapshot_bundle.tar" as the filename. Example: 20200605161704_snapshot_bundle.tar bootflash: <i>file-path</i>: The file path where the snapshot log file bundle is being stored on the bootflash. Choose one of the following initial paths: • bootflash:/// • bootflash://module-1/ • bootflash://sup-1/ • bootflash://sup-active/ • bootflash://sup-local/ logflash: <i>file-path</i>: The file path where the snapshot log file bundle is being stored on the logflash. Choose one of the following initial paths: • logflash:/// • logflash://module-1/ • logflash://sup-1/ • logflash://sup-active/ • logflash://sup-local/ usb1:: The file path where the snapshot log file bundle is being stored on the USB device. size <i>file-size</i>: The snapshot log file bundle based on size in megabytes (MB). Range is from 5MB through 250MB. time <i>minutes</i>: The snapshot log file bundle based on the last x amount of time (minutes). Range is from 1 minute through 30 minutes.

Example

```

switch# bloggerd log-snapshot snapshot1
Snapshot generated at logflash:evt_log_snapshot/snapshot1_snapshot_bundle.tar Please cleanup
once done.
switch#
switch# dir logflash:evt_log_snapshot
159098880 Dec 05 06:40:24 2019 snapshot1_snapshot_bundle.tar
159354880 Dec 05 06:40:40 2019 snapshot2_snapshot_bundle.tar

```

```
Usage for logflash://sup-local
759865344 bytes used
5697142784 bytes free
6457008128 bytes total
```

Display the same files using the command in this example:

```
switch# dir debug:log-snapshot-user/
159098880 Dec 05 06:40:24 2019 snapshot1_snapshot_bundle.tar
159354880 Dec 05 06:40:40 2019 snapshot2_snapshot_bundle.tar
```

```
Usage for debug://sup-local
929792 bytes used
4313088 bytes free
5242880 bytes total
```



Note Note the filename at the end of the example. Each individual log file is also identified by the date and time it was generated.

Beginning with Release 10.1(1), the LC core file includes the `log-snapshot` bundle. The `log-snapshot` bundle filename is `tac_snapshot_bundle.tar.gz`. An example is shown below:

```
bash-4.2$ tar -tvf 1610003655_0x102_aclqos_log.17194.tar.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 pss/
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_runtime_info_lc.gz
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_runtime_cfg_lc.gz
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_debug.gz
-rw-rw-rw- root/root 129583 2021-01-07 12:44 pss/clqosdb_ver1_0_user.gz
-rw-rw-rw- root/root 20291 2021-01-07 12:44 pss/clqosdb_ver1_0_node.gz
-rw-rw-rw- root/root 444 2021-01-07 12:44 pss/clqosdb_ver1_0_ctrl.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 proc/
-rw-rw-rw- root/root 15159 2021-01-07 12:44 0x102_aclqos_compress.17194.log.25162
-rw-rw-rw- root/root 9172392 2021-01-07 12:43 0x102_aclqos_core.17194.gz
-rw-rw-rw- root/root 43878 2021-01-07 12:44 0x102_aclqos_df_dmesg.17194.log.gz
-rw-rw-rw- root/root 93 2021-01-07 12:44 0x102_aclqos_log.17194
-rw-rw-rw- root/root 158 2021-01-07 12:44 0x102_aclqos_mcore.17194.log.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 usd17194/
-rw-rw-rw- root/root 11374171 2021-01-07 12:44 tac_snapshot_bundle.tar.gz
```

External Log File Storage

An external server solution provides the capability to store logs off-switch in a secure manner.

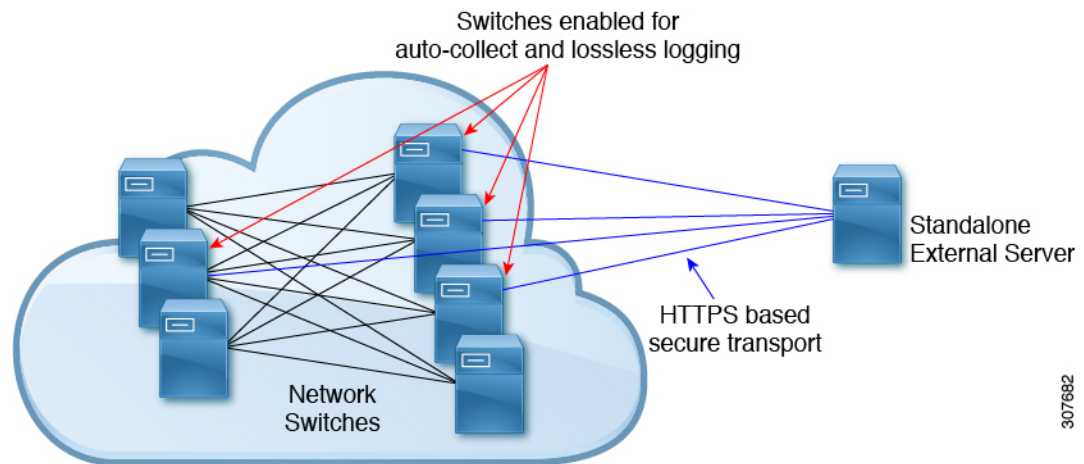


Note To create the external storage capability, contact Cisco Technical Assistance Center(TAC) to help deploy the external server solution.

The following are external log file storage capabilities:

- Enabled on-demand
- HTTPS-based transport
- Storage requirements:

- Nonmodular switches: 300MB
- Modular switches: 12GB (per day, per switch)
- An external server generally stores logs for 10 switches. However, there's no firm limit to the number of switches supported by an external server.



The external server solution has the following characteristics:

- Controller-less environment
- Manual management of security certificates
- Three supported use-cases:
 - Continuous collection of logs from selected switches
 - TAC-assisted effort to deploy and upload logs to Cisco servers.
 - Limited on-premise processing



Note Contact Cisco TAC for information regarding the setup and collection of log files in an external server.



CHAPTER 19

Configuring MAC Move Policy

This chapter describes how to configure the MAC move policy on Cisco NX-OS devices.

This chapter includes the following sections:

- [About MAC Move Policy, on page 337](#)
- [Guidelines and Limitations for MAC Move Policy, on page 337](#)
- [Configuring MAC Move Policy, on page 338](#)
- [Verifying the MAC Move Policy Configuration, on page 339](#)

About MAC Move Policy

The MAC move policy feature is used to monitor the movement of MAC addresses across the fabric. This feature provides the option to modify the MAC move parameters in order to enhance network stability and prevent the accumulation of Messaging and Transaction Service (MTS) in the Layer 2 Forwarder Manipulator (L2FM).

By default, the MAC move policy is disabled. For more information on configuring the MAC move policy and enabling the MTS buildup check, see [Configuring MAC Move Policy, on page 338](#) section.

The following are the advantages of MAC move policy:

- Enables tracking of MAC address movement at the MAC level to avoid loops.
- Checks periodically for MTS buildup to prevent crashes in the L2FM.
- Disables VLAN learning if the MAC move count surpasses the specified threshold.

Guidelines and Limitations for MAC Move Policy

MAC move Policy has the following configuration guidelines and limitations:

- Beginning with Cisco NX-OS Release 10.3(1)F, the MAC move policy is supported on Cisco Nexus 9300-X Cloud Scale switches.
- When the MAC move policy is enabled using the **mac-move policy** command, the default parameters (6 MAC moves in 30 seconds with a hold interval of 120 seconds) are set. You can modify the default set parameters. For more information, see [Configuring MAC Move Policy, on page 338](#).

- MAC move policy and the L2RIB detection cannot coexist with their default values. Both these mechanisms have a different approach to handle duplicate detection.
- If the MAC move policy is not required in a VXLAN environment, do not enable it. If it's required, make sure to modify either the L2RIB policy or the MAC move policy default value.
- L2RIB detection can be changed using the **l2rib dup-host-mac-detection <mac moves threshold> <detect-interval>** command.
- The following behavior is observed when both MAC move policy and L2RIB detection are configured:
 - When L2RIB detection is less than L2 policy, only L2RIB detection will be triggered. However, the L2 policy is not triggered.
 - When L2RIB detection is equal to L2 policy, either L2 policy or L2RIB detection will be triggered. However, there is no guarantee of the order in which these policies will be triggered.
 - When L2RIB detection is greater than L2 policy, only L2 policy will be triggered. However, the L2RIB detection is not triggered.

Configuring MAC Move Policy

This procedure enables/disables the MAC move policy on the switch:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] mac-move policy {[move-threshold<thresh>]} {[detect-intvl<d_intvl>]} {[hold-intvl<h_intvl>]} Example: <pre>switch(config)# mac-move policy move-threshold 6 detect-intvl 30 hold-intvl 120</pre>	Enables the MAC move policy. The option no disables the MAC move policy. • <i>thresh</i>: Maximum allowed moves. The range is 2–4096 and the default value is value 6. • <i>d_intvl</i>: Interval to track moves. The range is 30–3600 and the default value is 30. • <i>h_intvl</i>: Interval to disable learning. The range is 120–360 and the default value is 120.
Step 3	[no] mts-buildup check {[mts-percent<percent>]} {[detect-intvl<d_intvl>]}	Enables the mts-buildup check. The option no disables the mts-buildup check.

	Command or Action	Purpose
	Example: <pre>switch(config)# mts-buildup check mts-percent 40 detect-intvl 60</pre>	• <i>percent</i>: MTS buildup percent value. The range is 10–50 and the default value is 40. • <i>d_intvl</i>: Interval to check MTS buildup. The range is 60-600 and the default value is 60.

Verifying the MAC Move Policy Configuration

To display mac-move policy configuration information, perform one of the following tasks:

Command	Purpose
show mac-move policy	Displays information about the mac-move policy.
show mts-buildup check	Displays information about the mts-buildup check.

The following example shows sample output for the **show mac-move policy** command:

```
switch(config)# show mac-move policy
MAC move policy enabled = TRUE
MAC move policy threshold = 6
MAC move policy detect interval = 30
MAC move policy hold interval =120
switch(config)#
```

The following example shows sample output for the **show mts-buildup check** command when check is enabled:

```
switch(config)# show mts-buildup check
MTS buildup check enabled =TRUE
MTS check percent =40
MTS check interval =60
switch(config)#
```

The following example shows sample output for the **show mts-buildup check** command when check is disabled:

```
switch(config)# show mts-buildup check
MTS buildup check enabled =FALSE
ppwrks5(config)# show mac-move policy
MAC move policy enabled = FALSE
switch(config)#
```




CHAPTER 20

Terminal Lock for VSH Sessions

- [Terminal Lock for VSH Sessions, on page 341](#)

Terminal Lock for VSH Sessions

Overview

Currently with NX-OS, there are many users logged in to the switch and make configuration changes in their sessions with CLI. The goal is to restrict this scenario and allow only one user to configure the switch. This is achieved by terminal lock CLIs that lock the terminal to allow only one user to access the configure terminal commands. As a result, the effect of a “configuration lock” is achieved that prevents other users from changing the NX-OS running configuration.

Terminal lock feature provides a locking mechanism to enable users to have an exclusive configuration access to modify NX-OS running configuration.

The following is the sequence of operations:

1. `terminal lock` – This CLI provides the configuration lock to the user.
2. `terminal unlock` – This CLI releases the terminal lock taken by any session.
3. `show terminal lock` – This CLI shows the current terminal lock status and details.

Terminal Lock

The following are the guidelines for the terminal lock usage:

- `terminal lock` allows config commands to be executed only in that current session where the lock is held.
- `terminal lock` blocks only config commands in the other sessions, that means SHOW or EXEC CLIs are still allowed.
- Default timeout for terminal lock is 1800 seconds (30 minutes).
- Once the lock timer expires, terminal lock is released automatically.
- `terminal lock` CLI can be executed by any user with network-admin privilege.
- `terminal lock` is rejected if "configure dual-stage" session is in progress.

The following is the example CLI for the terminal lock:

```
switch# terminal lock?
lock Locks the CLI Config mode
switch# terminal lock ?
<CR>
<60-43200> Enter terminal lock timeout in seconds
*Default value is 1800
"terminal lock" locks the parser configuration mode and prints a syslog message as shown
in below example.
switch# terminal lock
switch# 2021 Jun 19 17:53:37 switch %VSHD-5-VSHD_CLI_TERM_LOCK: terminal lock is taken by
admin on console0
```



Note If a user tries to enter a configured terminal in another session, the following error message is displayed: "Configuration locked. terminal lock is taken by other VSH session."

Beginning with Cisco NX-OS Release 10.2(2)F, a new CLI option, "**terminal lock mdp**" is introduced to lock Model Driven Programmability interfaces like RESTCONF, NETCONF, gRPC, gNMI, and so on.

The CLI "**terminal lock mdp**" makes the terminal lock applicable to all configuration sessions including DME sessions.

The following is the sample output for the "**terminal lock mdp**" CLI:

```
switch# terminal lock?
lock Locks the CLI Config mode

switch# terminal lock ?
<CR>
<mdp> Locks Model Driven Programmability sessions
<60-43200> Enter terminal lock timeout in seconds
*Default value is 1800

switch# terminal lock mdp
2021 Oct 26 06:33:19 switch %VSHD-5-VSHD_CLI_TERM_LOCK: terminal lock is taken by admin on
console0
switch#
switch# show terminal lock
PID: 10018
User: admin
Session: console0
State: LOCKED
MDP lock: True
Lock acquired time: Mon Mar 8 09:24:03 2021
Lock Expiration timer (in Sec): 1800
switch#
```

Terminal Unlock

The following is the example CLI for the terminal unlock:

```
switch# terminal unlock?
unlock Force unlocking of the CLI config mode
switch# terminal unlock ?
<CR>
switch# terminal unlock
switch# 2021 Jun 19 17:53:21 switch %VSHD-5-VSHD_CLI_TERM_LOCK: terminal lock is released
by admin on console0
```



Note "terminal lock" can be taken by only one admin user, but lock can be released by any admin user using "terminal unlock."

Show Terminal Lock

This command displays the status and details of any current configuration locks, including the owner, user, session, lock state, and lock timer.

The following is the example CLI for the Show Terminal Lock when the lock is active:

```
switch# terminal lock
switch#
switch# show terminal lock
PID: 10018
User: admin
Session: console0
State: LOCKED
Lock acquired time: Mon Mar 8 09:24:03 2021
```

The following is the example CLI for the Show Terminal Lock when the lock is free:

```
switch# terminal unlock
switch#
switch#
switch# show terminal lock
PID: -1
User: unknown
Session: NA
State: FREE
Lock acquired time:
Lock Expiration timer (in Sec): 0
switch#
```




CHAPTER 21

Configuring Onboard Failure Logging

This chapter describes how to configure the onboard failure logging (OBFL) features on Cisco NX-OS devices.

This chapter includes the following sections:

- [About OBFL, on page 345](#)
- [Prerequisites for OBFL, on page 346](#)
- [Guidelines and Limitations for OBFL, on page 346](#)
- [Default Settings for OBFL, on page 346](#)
- [Configuring OBFL, on page 346](#)
- [Verifying the OBFL Configuration, on page 349](#)
- [Configuration Example for OBFL, on page 350](#)
- [Additional References, on page 350](#)

About OBFL

Cisco NX-OS provides the ability to log failure data to persistent storage, which you can retrieve and display for analysis at a later time. This onboard failure logging (OBFL) feature stores failure and environmental information in nonvolatile memory on the module. The information will help analyze failed modules.

OBFL stores the following types of data:

- Time of initial power-on
- Slot number of the module in the chassis
- Initial temperature of the module
- Firmware, BIOS, FPGA, and ASIC versions
- Serial number of the module
- Stack trace for crashes
- CPU hog information
- Memory leak information
- Software error messages
- Hardware exception logs

- Environmental history
- OBFL-specific history information
- ASIC interrupt and error statistics history
- ASIC register dumps

Prerequisites for OBFL

You must have network-admin user privileges.

Guidelines and Limitations for OBFL

OBFL has the following guidelines and limitations:

- OBFL is enabled by default.
- OBFL flash supports a limited number of writes and erases. The more logging that you enable, the faster you use up this number of writes and erases.



Note Be aware that the Cisco NX-OS commands for this feature may differ from those commands that are used in Cisco IOS.

Default Settings for OBFL

The following table lists the default settings for OBFL parameters.

Parameters	Default
OBFL	All features enabled

Configuring OBFL

You can configure the OBFL features on Cisco NX-OS devices.

Before you begin

Make sure that you are in global configuration mode.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	hw-module logging onboard Example: <pre>switch(config)# hw-module logging onboard Module: 7 Enabling ... was successful. Module: 10 Enabling ... was successful. Module: 12 Enabling ... was successful.</pre>	Enables all OBFL features.
Step 3	hw-module logging onboard counter-stats Example: <pre>switch(config)# hw-module logging onboard counter-stats Module: 7 Enabling counter-stats ... was successful. Module: 10 Enabling counter-stats ... was successful. Module: 12 Enabling counter-stats ... was successful.</pre>	Enables the OBFL counter statistics.
Step 4	hw-module logging onboard cpuhog Example: <pre>switch(config)# hw-module logging onboard cpuhog Module: 7 Enabling cpu-hog ... was successful. Module: 10 Enabling cpu-hog ... was successful. Module: 12 Enabling cpu-hog ... was successful.</pre>	Enables the OBFL CPU hog events.
Step 5	hw-module logging onboard environmental-history Example: <pre>switch(config)# hw-module logging onboard environmental-history Module: 7 Enabling environmental-history ... was successful. Module: 10 Enabling environmental-history ... was successful. Module: 12 Enabling environmental-history ... was successful.</pre>	Enables the OBFL environmental history.

	Command or Action	Purpose
Step 6	hw-module logging onboard error-stats Example: <pre>switch(config)# hw-module logging onboard error-stats Module: 7 Enabling error-stats ... was successful. Module: 10 Enabling error-stats ... was successful. Module: 12 Enabling error-stats ... was successful.</pre>	Enables the OBFL error statistics.
Step 7	hw-module logging onboard interrupt-stats Example: <pre>switch(config)# hw-module logging onboard interrupt-stats Module: 7 Enabling interrupt-stats ... was successful. Module: 10 Enabling interrupt-stats ... was successful. Module: 12 Enabling interrupt-stats ... was successful.</pre>	Enables the OBFL interrupt statistics.
Step 8	hw-module logging onboard module slot Example: <pre>switch(config)# hw-module logging onboard module 7 Module: 7 Enabling ... was successful.</pre>	Enables the OBFL information for a module.
Step 9	hw-module logging onboard obfl-logs Example: <pre>switch(config)# hw-module logging onboard obfl-logs Module: 7 Enabling obfl-log ... was successful. Module: 10 Enabling obfl-log ... was successful. Module: 12 Enabling obfl-log ... was successful.</pre>	Enables the boot uptime, device version, and OBFL history.
Step 10	(Optional) show logging onboard Example: <pre>switch(config)# show logging onboard</pre>	Displays information about OBFL. Note To display OBFL information stored in flash on a module, see Verifying the OBFL Configuration, on page 349 .
Step 11	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Verifying the OBFL Configuration

To display OBFL information stored in flash on a module, perform one of the following tasks:

Command	Purpose
show logging onboard boot-uptime	Displays the boot and uptime information.
show logging onboard counter-stats	Displays statistics on all ASIC counters.
show logging onboard credit-loss	Displays OBFL credit loss logs.
show logging onboard device-version	Displays device version information.
show logging onboard endtime	Displays OBFL logs to a specified end time.
show logging onboard environmental-history	Displays environmental history.
show logging onboard error-stats	Displays error statistics.
show logging onboard exception-log	Displays exception log information.
show logging onboard interrupt-stats	Displays interrupt statistics.
show logging onboard module <i>slot</i> internal reset-reason	Displays OBFL information for a specific module. Note If you specify internal reset-reason and you are operating in a redundant supervisor configuration, checking the persistent log on the standby supervisor after a system reset occurs will display a relevant reset reason. The reset reason is recorded on the on-board flash for both the active and standby supervisor.
show logging onboard obfl-history	Displays history information.
show logging onboard obfl-logs	Displays log information.
show logging onboard stack-trace	Displays kernel stack trace information.
show logging onboard starttime	Displays OBFL logs from a specified start time.
show logging onboard status	Displays OBFL status information.

Use the **show logging onboard status** command to display the configuration status of OBFL.

```
switch# show logging onboard status
-----
OBFL Status
-----
Switch OBFL Log: Enabled

Module: 4 OBFL Log: Enabled
cpu-hog Enabled
credit-loss Enabled
environmental-history Enabled
error-stats Enabled
exception-log Enabled
interrupt-stats Enabled
mem-leak Enabled
```

```

miscellaneous-error Enabled
obfl-log (boot-uptime/device-version/obfl-history) Enabled
register-log Enabled
request-timeout Enabled
stack-trace Enabled
system-health Enabled
timeout-drops Enabled
stack-trace Enabled

Module: 22 OBFL Log: Enabled
cpu-hog Enabled
credit-loss Enabled
environmental-history Enabled
error-stats Enabled
exception-log Enabled
interrupt-stats Enabled
mem-leak Enabled
miscellaneous-error Enabled
obfl-log (boot-uptime/device-version/obfl-history) Enabled
register-log Enabled
request-timeout Enabled
stack-trace Enabled
system-health Enabled
timeout-drops Enabled
stack-trace Enabled

```

Use the **clear logging onboard** command to clear the OBFL information for each of the **show** command options listed.

Configuration Example for OBFL

This example shows how to enable OBFL on module 2 for environmental information:

```

switch# configure terminal
switch(config)# hw-module logging onboard module 2 environmental-history

```

Additional References

Related Documents

Related Topic	Document Title
Configuration files	<i>Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide</i>



CHAPTER 22

Switched Port Analyzer

A switched port analyzer (SPAN) is a network monitoring feature to analyze traffic between ports on Cisco NX-OS devices by directing the SPAN session traffic to a destination port with an external analyzer attached to it. You can define the sources and destinations to monitor in a SPAN session on the local device.

- [About SPAN, on page 351](#)
- [Prerequisites, on page 355](#)
- [Guidelines and Limitations for SPAN, on page 356](#)
- [Default Settings, on page 365](#)
- [Configuring SPAN, on page 366](#)
- [Verifying the SPAN Configuration, on page 377](#)
- [Configuration Examples for SPAN, on page 378](#)
- [Additional References, on page 382](#)

About SPAN

SPAN analyzes all traffic between source ports by directing the SPAN session traffic to a destination port with an external analyzer attached to it.

You can define the sources and destinations to monitor in a SPAN session on the local device.

Ports

Sources

The interfaces from which traffic can be monitored are called SPAN sources. Sources designate the traffic to monitor and whether to copy ingress (Rx), egress (Tx), or both directions of traffic.

SPAN sources include the following:

- Ethernet ports (but not subinterfaces)
- The inband interface to the control plane CPU



Note When you specify the supervisor inband interface as a SPAN source, the device monitors all packets that are sent by the Supervisor CPU.

- VLANs

- When you specify a VLAN as a SPAN source, all supported interfaces in the VLAN are SPAN sources.
- VLANs can be SPAN sources only in the ingress direction.

**Note**

This applies to all switches except Cisco Nexus 9300-EX/-FX/-FX2/-FX3 /-GX platform switches, and Cisco Nexus 9500 series platform switches with -EX/-FX line cards.

- Satellite ports and host interface port channels on the Cisco Nexus 2000 Series Fabric Extender (FEX)
 - These interfaces are supported in Layer 2 access mode and Layer 2 trunk mode. They are not supported in Layer 3 mode, and Layer 3 subinterfaces are not supported.
 - Cisco Nexus 9300 and 9500 platform switches support FEX ports as SPAN sources in the ingress direction for all traffic and in the egress direction only for known Layer 2 unicast traffic flows through the switch and FEX. Routed traffic might not be seen on FEX HIF egress SPAN.

SPAN source ports have the following characteristics:

- A port configured as a source port cannot also be configured as a destination port.
- If you use the supervisor inband interface as a SPAN source, all packets generated by the supervisor hardware (egress) are monitored.

**Note**

Rx is from the perspective of the ASIC (traffic egresses from the supervisor over the inband and is received by the ASIC/SPAN).

**Note**

A single SPAN session can include mixed sources in any combination of the above.

Destinations

SPAN destinations refer to the interfaces that monitor source ports. Destination ports receive the copied traffic from SPAN sources. SPAN destinations include the following:

- Ethernet ports
- Port channels
- CPU as destination port
- Uplink ports on Cisco Nexus 9300 Series switches



Note FEX ports are not supported as SPAN destination ports.

SPAN destination ports have the following characteristics:

- A port configured as a destination port cannot also be configured as a source port.
- The same destination interface cannot be used for multiple SPAN sessions.
- Destination ports do not participate in any spanning tree instance. SPAN output includes bridge protocol data unit (BPDU) Spanning Tree Protocol hello packets.

Characteristics of Source Ports

SPAN source ports have the following characteristics:

- A port configured as a source port cannot also be configured as a destination port.
- If you use the supervisor inband interface as a SPAN source, all packets generated by the supervisor hardware (egress) are monitored.



Note Rx is from the perspective of the ASIC (traffic egresses from the supervisor over the inband and is received by the ASIC/SPAN).

SPAN Destination Ports

SPAN destinations refer to the interfaces that monitor source ports. Destination ports receive the copied traffic from SPAN sources. SPAN destinations include the following:

- Ethernet ports
- Port channels
- CPU as destination port
- Uplink ports on Cisco Nexus 9300 Series switches



Note FEX ports are not supported as SPAN destination ports.

Characteristics

SPAN destination ports have the following characteristics:

- A port configured as a destination port cannot also be configured as a source port.
- The same destination interface cannot be used for multiple SPAN sessions.
- Destination ports do not participate in any spanning tree instance. SPAN output includes bridge protocol data unit (BPDU) Spanning Tree Protocol hello packets.

Characteristics of Destination Ports

SPAN destination ports have the following characteristics:

- A port configured as a destination port cannot also be configured as a source port.
- The same destination interface cannot be used for multiple SPAN sessions.
- Destination ports do not participate in any spanning tree instance. SPAN output includes bridge protocol data unit (BPDU) Spanning Tree Protocol hello packets.

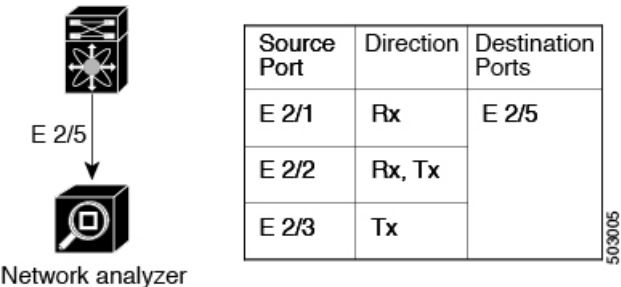
Sessions

You can create SPAN sessions to designate sources and destinations to monitor.

See the *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide* for information on the number of supported SPAN sessions.

This figure shows a SPAN configuration. Packets on three Ethernet ports are copied to destination port Ethernet 2/5. Only traffic in the direction specified is copied.

Figure 7: SPAN Configuration



-
-
-

Localized SPAN Sessions

A SPAN session is localized when all of the source interfaces are on the same line card. A session destination interface can be on any line card.



Note A SPAN session with a VLAN source is not localized.

Localized SPAN Sessions

A SPAN session is localized when all of the source interfaces are on the same line card. A session destination interface can be on any line card.



Note A SPAN session with a VLAN source is not localized.

SPAN Functionality

-
-
-

SPAN Truncation

Beginning with Cisco NX-OS Release 7.0(3)I7(1), you can configure the truncation of source packets for each SPAN session based on the size of the MTU. Truncation helps to decrease SPAN bandwidth by reducing the size of monitored packets. Any SPAN packet that is larger than the configured MTU size is truncated to the given size. For example, if you configure the MTU as 300 bytes, the packets with greater than 300 bytes are truncated to 300 bytes.

SPAN truncation is disabled by default. To use truncation, you must enable it for each SPAN session.

ACL TCAM Regions

You can change the size of the ACL ternary content addressable memory (TCAM) regions in the hardware. For information on the TCAM regions used by SPAN sessions, see the Configuring IP ACLs chapter of the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

High Availability

The SPAN feature supports stateless and stateful restarts. After a reboot or supervisor switchover, the running configuration is applied. For more information on high availability, see the [Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide](#).

ACL TCAM Regions

You can change the size of the ACL ternary content addressable memory (TCAM) regions in the hardware. For information on the TCAM regions used by SPAN sessions, see the Configuring IP ACLs chapter of the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

High Availability

The SPAN feature supports stateless and stateful restarts. After a reboot or supervisor switchover, the running configuration is applied. For more information on high availability, see the [Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide](#).

Prerequisites

SPAN has the following prerequisites:

- You must first configure the ports on each device to support the desired SPAN configuration. For more information, see the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide.

Guidelines and Limitations for SPAN



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

SPAN has the following configuration guidelines and limitations:

- A maximum of 48 source interfaces are supported per SPAN session (Rx and Tx, Rx, or Tx).
- Traffic that is denied by an ACL may still reach the SPAN destination port because SPAN replication is performed on the ingress side prior to the ACL enforcement (ACL dropping traffic).
- For SPAN session limits, see the *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.
- You can configure maximum of 32 source VLANs while configuring SPAN session.
- All SPAN replication is performed in the hardware. The supervisor CPU is not involved.
- You can configure a SPAN session on the local device only.
- Configuring two SPAN or ERSPAN sessions on the same source interface with only one filter is not supported. If the same source is used in multiple SPAN or ERSPAN sessions, either all the sessions must have different filters or no sessions should have filters.
- Packets with FCS errors are not mirrored in a SPAN session.
- The following guidelines apply to SPAN copies of access port dot1q headers:
 - When traffic ingresses from a trunk port or a routed port and egresses to an access port, an egress SPAN copy of an access port on a switch interface always has a dot1q header.
 - When traffic ingresses from an access port and egresses to a trunk port or a routed port, an ingress SPAN copy of an access port on a switch interface does not have a dot1q header.
 - When traffic ingresses from an access port and egresses to an access port, an ingress/egress SPAN copy of an access port on a switch interface does not have a dot1q header.
 -
- You can configure only one destination port in a SPAN session.
- SPAN mirroring is not supported for PBR traffic.
- You cannot configure a port as both a source and destination port.
- Enabling UniDirectional Link Detection (UDLD) on the SPAN source and destination ports simultaneously is not supported. If UDLD frames are expected to be captured on the source port of such SPAN session, disable UDLD on the destination port of the SPAN session.
- SPAN is not supported for management ports.
- Statistics are not support for the filter access group.

- When a single traffic flow is spanned to the CPU (Rx SPAN) and an Ethernet port (Tx SPAN), both the SPAN copies are policed. Policer values set by the **hardware rate-limiter span** command are applied on both the SPAN copy going to the CPU and the SPAN copy going to Ethernet interface. This limitation applies to the following switches:
 - Cisco Nexus 92348GC-X, Cisco Nexus 9332C, and Cisco Nexus 9364C switches
 - Cisco Nexus 9300-EX, FX, FX2, FX3, GX platform switches
 - Cisco Nexus 9504, 9508, and 9516 platform switches with EX and FX line cards
- SPAN is supported in Layer 3 mode; however, SPAN is not supported on Layer 3 subinterfaces or Layer 3 port-channel subinterfaces.
- When a SPAN session contains source ports that are monitored in the transmit or transmit and receive direction, packets that these ports receive can be replicated to the SPAN destination port although the packets are not actually transmitted on the source ports. Some examples of this behavior on source ports are as follows:
 - Traffic that results from flooding
 - Broadcast and multicast traffic
- SPAN sessions cannot capture packets with broadcast or multicast MAC addresses that reach the supervisor, such as ARP requests and Open Shortest Path First (OSPF) protocol hello packets, if the source of the session is the supervisor Ethernet in-band interface. To capture these packets, you must use the physical interface as the source in the SPAN sessions.
- VLAN SPAN monitors only the traffic that enters Layer 2 ports in the VLAN.
- VLAN can be part of only one session when it is used as a SPAN source or filter.
- VLAN ACL redirects to SPAN destination ports are not supported.
- When using a VLAN ACL to filter a SPAN, only **action forward** is supported; **action drop** and **action redirect** are not supported.
- The combination of VLAN source session and port source session is not supported. If the traffic stream matches the VLAN source session and port source session, two copies are needed at two destination ports. Due to the hardware limitation, only the VLAN source SPAN and the specific destination port receive the SPAN packets. This limitation applies only to the following Cisco devices:

Table 19: Cisco Nexus 9000 Series Switches

Cisco Nexus 93120TX	Cisco Nexus 93128TX	Cisco Nexus 9332PQ
Cisco Nexus 9372PX	Cisco Nexus 9372PX-E	Cisco Nexus 9372TX
Cisco Nexus 9396PX	Cisco Nexus 9372TX-E	Cisco Nexus 9396TX

Table 20: Cisco Nexus 9000 Series Line Cards, Fabric Modules, and GEM Modules

N9K-X9408PC-CFP2	N9K-X9536PQ	N9K-C9504-FM
N9K-X9432PQ	N9K-X9464TX	—

- When you filter a monitor session, make sure that the access-group specified must be a VACL, or VLAN access-map and not a regular ACL for filtering purpose. This guideline is not applicable for Cisco Nexus 9508 switches with 9636C-R and 9636Q-R line cards.
- An access-group filter in a SPAN session must be configured as vlan-accessmap. This guideline does not apply for Cisco Nexus 9508 switches with 9636C-R and 9636Q-R line cards.
- Supervisor-generated stream of bytes module header (SOBMH) packets have all the information to go out on an interface and can bypass all forwarding lookups in the hardware, including SPAN and ERSPAN. CPU-generated frames for Layer 3 interfaces and the Bridge Protocol Data Unit (BPDU) class of packets are sent using SOBMH. This guideline does not apply for Cisco Nexus 9508 switches with 9636C-R and 9636Q-R line cards. The Cisco Nexus 9636C-R and 9636Q-R both support inband SPAN and local SPAN.
- Cisco NX-OS does not span Link Layer Discovery Protocol (LLDP) or Link Aggregation Control Protocol (LACP) packets when the source interface is not a host interface port channel.
- SPAN copies for multicast packets are made before rewrite. Therefore, the TTL, VLAN ID, any remarking due to an egress policy, and so on, are not captured in the SPAN copy.
- If SPAN is mirroring the traffic which ingresses on an interface in an ASIC instance and egresses on a Layer 3 interface (SPAN Source) on a different ASIC instance, then a Tx mirrored packet has a VLAN ID of 4095 on Cisco Nexus 9300 platform switches (except EX, FX, or FX2) and Cisco Nexus 9500 platform modular switches.
- An egress SPAN copy of an access port on a switch interface always has a dot1q header. This guideline does not apply for Cisco Nexus 9508 platform switches with 9636C-R and 9636Q-R line cards.
- The flows for post-routed unknown unicast flooded packets are in the SPAN session, even if the SPAN session is configured not to monitor the ports on which this flow is forwarded. This limitation applies to Network Forwarding Engine (NFE) and NFE2-enabled EOR switches and SPAN sessions that have Tx port sources.
- VLAN sources are spanned only in the Rx direction. This limitation does not apply to the following switch platforms which support VLAN spanning in both directions:
 - Cisco Nexus 9300-EX platform switches
 - Cisco Nexus 9300-FX platform switches
 - Cisco Nexus 9300-FX2 platform switches
 - Cisco Nexus 9300-FX3 platform switches
 - Cisco Nexus 9300-GX platform switches
 - Cisco Nexus 9504, 9508, and 9516 switches with the 97160YC-EX line card.
 - Cisco Nexus 9508 switches with 9636C-R and 9636Q-R line cards.
- If a VLAN source is configured as both directions in one session and the physical interface source is configured in two other sessions, Rx SPAN is not supported for the physical interface source session. This limitation applies to the Cisco Nexus 97160YC-EX line card.
- With regard to session filtering functionality, ACL filter is supported only in Rx source, and VLAN filter is supported in both Tx and Rx sources. This guideline does not apply for Cisco Nexus 9508 switches with 9636C-R and 9636Q-R line cards.

- Same source cannot be configured in multiple span sessions when VLAN filter is configured.
- The FEX NIF interfaces or port-channels cannot be used as a SPAN source or SPAN destination. If the FEX NIF interfaces or port-channels are specified as a SPAN source or SPAN destination, the software displays an unsupported error.
- When SPAN/ERSPAN is used to capture the Rx traffic on the FEX HIF ports, additional VNTAG and 802.1Q tags are present in the captured traffic.
- VLAN and ACL filters are not supported for FEX ports.
- If the sources used in bidirectional SPAN sessions are from the same FEX, the hardware resources are limited to two SPAN sessions.
- Truncation is supported only for local and ERSPAN source sessions. It is not supported for ERSPAN destination sessions.
- When sFlow is configured on N9K-C9508-FM-G with the N9K-X9716D-GX line card, disable sFlow before configuring SPAN sessions.
- Configuring MTU on a SPAN session truncates all packets egressing on the SPAN destination (for that session) to the MTU value specified.
 - The cyclic redundancy check (CRC) is recalculated for the truncated packet.
 - The bytes specified are retained starting from the header of the packets. The rest are truncated if the packet is longer than the MTU.
- Beginning with Cisco NX-OS Release 10.1(2), SPAN is supported on the Cisco N9K-X9624D-R2 line card.
- Beginning with Cisco NX-OS Release 10.2(1q)F, SPAN is supported on the N9K-C9332D-GX2B platform switches.
- MTU truncation is not supported on Cisco Nexus 9504/9508 modular chassis with the N9K-X9636C-R, N9K-X9636Q-R, N9K-X9636C-RX, and N9K-X96136YC-R line cards.
- Beginning with Cisco NX-OS Release 10.2(2)F, Multicast SPAN Tx is supported on Cisco Nexus 9300-GX, 9300-GX2, and 9300-FX3 platform switches.
- Beginning with Cisco NX-OS Release 10.3(1)F, SPAN is supported on Cisco Nexus 9808 platform switches.
- When configuring local SPAN sessions or ERSPAN-source monitor sessions with a filter access-group rule, we recommend to configure the necessary sub-commands of the VLAN access-map too, such as the match rule.

SPAN Limitations for the Cisco Nexus 3000 Platform Switches

The following guidelines and limitations apply only the Nexus 3000 Series switches running Cisco Nexus 9000 code:

- The Cisco Nexus 3232C and 3264Q switches do not support SPAN on CPU as destination.

SPAN Limitations for the Cisco Nexus 9200 Platform Switches (excluding 9232E-B1)



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

The following guidelines and limitations apply only the Cisco Nexus 9200 platform switches:

- For Cisco Nexus 9200 platform switches, Rx SPAN is not supported for multicast without a forwarding interface on the same slice as the SPAN destination port.
- Tx SPAN for multicast, unknown multicast, and broadcast traffic are not supported on the Cisco Nexus 9200 platform switches.
- Tx SPAN of CPU-generated packets is not supported on Cisco Nexus 9200 platform switches.
- UDF-based SPAN is supported on the Cisco Nexus 9200 platform switches.
- The Cisco Nexus 9200 platform switches do not support Multiple ACL filters on the same source.
- VLAN Tx SPAN is supported on the Cisco Nexus 9200 platform switches.
- When multiple egress ports on the same slice are congested by egressing SPAN traffic, those egress ports will not get the line rate on the Cisco Nexus 9200 platform switches.
- Using the ACL filter to span subinterface traffic on the parent interface is not supported on the Cisco Nexus 9200 platform switches.
- On the Cisco Nexus 9200 platform switches, the CPU SPAN source can be added only for the Rx direction (SPAN packets coming from the CPU).
- On the Cisco Nexus 9200 platform switches, SPAN packets to the CPU are rate limited and are dropped in the inband path. You can change the rate limit using the **hardware rate-limiter span** command. You can analyze SPAN copies on the supervisor using the **ethanalyzer local interface inband mirror detail** command.

SPAN Limitations for the Cisco Nexus 9300 Platform Switches



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

The following guidelines and limitations apply only the Cisco Nexus 9300 platform switches:

- SPAN does not support ECMP hashing/load balancing at the source on Cisco Nexus 9300-GX platform switches.
- The following filtering limitations apply to egress (Tx) SPAN on all Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches:
 - ACL filtering is not supported (applies to both unicast and Broadcast, Unknown Unicast and Multicast (BUM) traffic)

- VLAN filtering is supported, but only for unicast traffic
- VLAN filtering is not supported for BUM traffic
- Cisco Nexus 9300-FX2 platform switches support simultaneous SPAN and sFlow functionality.
- Cisco Nexus 9300-EX/FX platform switches do not support simultaneous SPAN and sFlow functionality.
- Cisco Nexus 9300-EX/FX/FX2 platform switches support simultaneous SPAN and NetFlow functionality.
- VLAN Tx SPAN is supported on Cisco Nexus 9300-EX and FX platform switches
- Cisco Nexus 9300 platform switches support multiple ACL filters on the same source.
- A single forwarding engine instance supports four SPAN sessions. For Cisco Nexus 9300 platform switches, if the first three sessions have bidirectional sources, the fourth session has hardware resources only for Rx sources.
- Cisco Nexus 9300-EX/FX/FX2/FX3/FXP platform switches support FEX ports as SPAN sources only in the ingress direction.
- Cisco Nexus 9300 platform switches (excluding Cisco Nexus 9300-EX/FX/FX2/FX3/FXP switches) support FEX ports as SPAN sources in the ingress direction for all traffic and in the egress direction only for known Layer 2 unicast traffic flows through the switch and FEX. Routed traffic might not be seen on FEX HIF egress SPAN.
- Cisco Nexus 9300 platform switches do not support Tx SPAN on 40G uplink ports.

**Note**

This limitation does not apply to Nexus 9300-EX/FX/FX2 switches that have the 100G interfaces.

- Tx SPAN of CPU-generated packets is not supported on Cisco Nexus 9200, 9300-EX/FX/FXP/FX2/FX3/GX/GX2, 9300C, C9516-FM-E2, and C9508-FM-E2 switches.
- Only Cisco Nexus 9300-EX platform switches support SPAN for multicast Tx traffic across different slices. The slices must be on the same leaf spine engine (LSE).
- For Tx interface SPAN with Layer 2 switch port and port-channel sources on Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches, only one copy is made per receiver unit regardless of how many Layer 2 members are receiving the stream in the same VLAN. For example, if e1/1-8 are all Tx direction SPAN sources and all are joined to the same group, the SPAN destination port sees one pre-rewrite copy of the stream, not eight copies. In addition, if for any reason one or more of those ports drops the packets on egress (for example, due to congestion), the packets may still reach the SPAN destination port. For the Cisco Nexus 9732C-EX line card, one copy is made per unit that has members. For port-channel sources, the Layer 2 member that will SPAN is the first port-channel member.
- SPAN Tx broadcast and SPAN Tx multicast are supported for Layer 2 port and port-channel sources across slices on Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches, and the Cisco Nexus 9732C-EX line card, but only when IGMP snooping is disabled. (Otherwise, the slice limitation still applies.) These features are not supported for Layer 3 port sources, FEX ports (with unicast or multicast traffic), and VLAN sources.

- For SPAN Tx multicast for Layer 2, SPAN copies are created independent of multicast replication. Due to this, multicast and SPAN packet have different values for VLAN tag, which is the ingress interface VLAN ID.
- A SPAN copy of Cisco Nexus 9300 platform switch 40G uplink interfaces will miss the dot1q information when spanned in the Rx direction.



Note This limitation does not apply to Nexus 9300-EX/FX/FX2 platform switches that have the 100G interfaces.

- UDF-based SPAN is supported on the Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches.
- UDF-SPAN acl-filtering only supports source interface rx. This limitation applies to the following switches:
 - Cisco Nexus 9332PQ
 - Cisco Nexus 9372PX
 - Cisco Nexus 9372PX-E
 - Cisco Nexus 9372TX
 - Cisco Nexus 9372TX-E
 - Cisco Nexus 93120TX
- The Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches do not support Multiple ACL filters on the same source.
- When multiple egress ports on the same slice are congested by egressing SPAN traffic, those egress ports will not get the line rate on the Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches.
- Using the ACL filter to span subinterface traffic on the parent interface is not supported on the Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches.
- On the Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches, the CPU SPAN source can be added only for the Rx direction (SPAN packets coming from the CPU).
- On the Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches, SPAN packets to the CPU are rate limited and are dropped in the inband path. You can change the rate limit using the **hardware rate-limiter span** command. You can analyze SPAN copies on the supervisor using the **ethanalyzer local interface inband mirror detail** command.
- The following Cisco Nexus switches support sFlow and SPAN together:
 - Cisco Nexus 9336C-FX2
 - Cisco Nexus 93240YC-FX2
 - Cisco Nexus 93360YC-FX2
- Beginning with Cisco NX-OS Release 9.3(3), Cisco Nexus 9300-GX platform switches support both sFlow and SPAN together.

- Beginning with Cisco NX-OS Release 9.3(5), Cisco Nexus 9300-GX platform switches support SPAN truncation.
- Beginning with Cisco NX-OS Release 10.1(1), sFlow and SPAN are supported on Cisco N9K-C93180YC-FX3 platform switches.
- Beginning from Cisco NX-OS Release 10.2(3)F, the FC span feature provides packet capture support for FC ports, SAN port channels, and VSANs for both NPV and SAN switching modes on Cisco Nexus C93180YC-FX, C9336C-FX2-E, and C93360YC-FX2 platform switches.
- FC ports, SAN Port channel, and VSANs as source are not supported in ERSPAN.
- FC ports, SAN Port channel, and VSANs cannot be added as source in more than one span sessions.
- The guideline—A single forwarding engine instance supports four active SPAN sessions—is also applicable to the FC span feature.
- SNMP support for FC span feature is not available in Cisco NX-OS Release 10.2(3)F.

SPAN Limitations for the Cisco Nexus 9500 Platform Switches



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

The following guidelines and limitations apply only the Cisco Nexus 9500 platform switches:

- The following filtering limitations apply to egress (Tx) SPAN on 9500 platform switches with EX or FX line cards:
 - ACL filtering is not supported (applies to both unicast and Broadcast, Unknown Unicast and Multicast (BUM) traffic)
 - VLAN filtering is supported, but only for unicast traffic
 - VLAN filtering is not supported for BUM traffic
- FEX and SPAN port-channel destinations are not supported on the Cisco Nexus 9500 platform switches with EX or FX line cards.
- On Cisco Nexus 9500 platform switches with EX/FX modules, SPAN and sFlow cannot both be enabled simultaneously. If one is active, the other cannot be enabled. However, on the Cisco Nexus 9500 platform switches with EX or FX line cards, NetFlow and SPAN can both be enabled simultaneously, providing a viable alternative to using sFlow and SPAN.
- Cisco Nexus 9500 platform switches support VLAN Tx SPAN with the following line cards:
 - Cisco Nexus 97160YC-EX
 - Cisco Nexus 9732C-EX
 - Cisco Nexus 9732C-FX
 - Cisco Nexus 9736C-EX
 - Cisco Nexus 9736C-FX

- Cisco Nexus 9736Q-FX
- Cisco Nexus 9788TC-FX
- Cisco Nexus 9500 platform switches support multiple ACL filters on the same source.
- Tx SPAN of CPU-generated packets is not supported on Cisco Nexus 9500 platform switches with EX-based line cards.
- TCAM carving is not required for SPAN/ERSPAN on the following line cards:
 - Cisco Nexus 9636C-R
 - Cisco Nexus 9636Q-R
 - Cisco Nexus 9636C-RX
 - Cisco Nexus 96136YC-R
 - Cisco Nexus 9624D-R2



Note All other switches supporting SPAN/ERSPAN must use TCAM carving.

- On the Cisco Nexus 9500 platform switches, depending on the SPAN source's forwarding engine instance mappings, a single forwarding engine instance may support four SPAN sessions. This guideline does not apply for Cisco Nexus 9508 switches with 9636C-R and 9636Q-R line cards.
- Same source interface cannot be configured in multiple SPAN sessions on N9K-X96136YC-R line card.
- Multiple ACL filters are not supported on the same source.
- Cisco Nexus 9500 platform switches support FEX ports as SPAN sources in the ingress direction for all traffic and in the egress direction only for known Layer 2 unicast traffic flows through the switch and FEX. Routed traffic might not be seen on FEX HIF egress SPAN.
- SPAN does not support destinations on Cisco Nexus 9408PC-CFP2 line card ports.
- Truncation is supported for Cisco Nexus 9500 platform switches with N9K-X97160YC-EX 9700-EX or 9700-FX line cards.
- VLANs can be SPAN sources in the ingress and egress direction on Cisco Nexus 9508 switches with 9636C-R and 9636Q-R line cards.
- UDF-SPAN acl-filtering only supports source interface rx. This limitation applies to the following line cards:
 - Cisco Nexus 9564PX
 - Cisco Nexus 9464TX2
 - Cisco Nexus 9464TX
 - Cisco Nexus 9464TX2
 - Cisco Nexus 9564TX
 - Cisco Nexus 9464PX

- Cisco Nexus 9536PQ
- Cisco Nexus 9636PQ
- Cisco Nexus 9432PQ

SPAN Guidelines and Limitations for Cisco Nexus 9800 and 9232E-B1 Platform Switches



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide* on [Cisco.com](https://www.cisco.com).

The following guidelines and limitations apply only the Cisco Nexus 9800 and 9232-B1 platform switches:

- Only RX is supported on SPAN to CPU.
- Sharing of the same source port or interface across sessions is not supported.
- A maximum of 10 monitor sessions are supported at a time.
- 10 active SPAN sessions are supported at a time.
- Monitor stats are not displayed for SPAN to CPU.
- SPAN is not supported on Layer 2 ports, port-channel, and tunnel ports.
- VLAN as source is not supported on SPAN.
- MTU truncation is only supported for 343 bytes.
- MTU truncation is supported only on RX and not on TX.
- UDF is not supported.
- SPAN is not supported on subinterfaces.
- Due to SDK limitations, mirrored multicast traffic is processed as unicast traffic instead of multicast traffic.

Default Settings

The following table lists the default settings for SPAN parameters.

Parameters	Default
SPAN sessions	Created in the shut state

Configuring SPAN



Note Cisco NX-OS commands for this feature may differ from those in Cisco IOS.

Configuring a SPAN Session

You can configure a SPAN session on the local device only. By default, SPAN sessions are created in the shut state.



Note For bidirectional traditional sessions, you can configure the sessions without specifying the direction of the traffic.

Before you begin

You must configure the destination ports in access or trunk mode. For more information, see the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface ethernet slot/port Example: <pre>switch(config)# interface ethernet 2/5 switch(config-if)#</pre>	Enters interface configuration mode on the selected slot and port.
Step 3	switchport Example: <pre>switch(config-if)# switchport</pre>	Configures switchport parameters for the selected slot and port or range of ports.
Step 4	switchport monitor Example: <pre>switch(config-if)# switchport monitor</pre>	Configures the switchport interface as a SPAN destination.
Step 5	(Optional) Repeat Steps 2 through 4 to configure monitoring on additional SPAN destinations.	—

	Command or Action	Purpose
Step 6	no monitor session <i>session-number</i> Example: <pre>switch(config)# no monitor session 3</pre>	Clears the configuration of the specified SPAN session. The new session configuration is added to the existing session configuration.
Step 7	monitor session <i>session-number</i> [rx tx] [shut] Example: <pre>switch(config)# monitor session 3 rx switch(config-monitor)#</pre> Example: <pre>switch(config)# monitor session 3 tx switch(config-monitor)#</pre> Example: <pre>switch(config)# monitor session 3 shut switch(config-monitor)#</pre>	Enters the monitor configuration mode. The new session configuration is added to the existing session configuration. By default, the session is created in the shut state, and the session is a local SPAN session. The optional keyword shut specifies a shut state for the selected session.
Step 8	description <i>description</i> Example: <pre>switch(config-monitor)# description my_span_session_3</pre>	Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters.
Step 9	source { interface type [rx tx both] [vlan { <i>number</i> <i>range</i> } [rx]} [vsan { <i>number</i> <i>range</i> } [rx]} Example: <pre>switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx</pre> Example: <pre>switch(config-monitor)# source interface fc1/1 both</pre> Example: <pre>switch(config-monitor)# source interface port-channel 2</pre> Example: <pre>switch(config-monitor)# source interface san-port-channel201 both</pre> Example: <pre>switch(config-monitor)# source interface sup-eth 0 rx</pre> Example: <pre>switch(config-monitor)# source vlan 3, 6-8 rx</pre> Example:	Configures sources and the traffic direction in which to copy packets. You can enter a range of Ethernet ports, FC ports, a port channel, SAN port channels, an inband interface, a range of VLANs, a range of VSANs, or a satellite port or host interface port channel on the Cisco Nexus 2000 Series Fabric Extender (FEX). You can configure one or more sources, as either a series of comma-separated entries or a range of numbers. You can specify the traffic direction to copy as ingress (rx), egress (tx), or both. Note Source VLANs are supported only in the ingress direction. Source FEX ports are supported in the ingress direction for all traffic and in the egress direction only for known Layer 2 unicast traffic. This note does not apply to Cisco Nexus 9300-EX/-FX/-FX2/-FX3/-GX series platform switches, and Cisco Nexus 9500 series platform switches with -EX/-FX line cards.

	Command or Action	Purpose
	<pre>switch(config-monitor)# source vsan 500 rx</pre> Example: <pre>switch(config-monitor)# source interface ethernet 101/1/1-3</pre>	Supervisor as a source is only supported in the Rx direction. For a unidirectional session, the direction of the source must match the direction specified in the session. Note Source VSANs are also supported only in the ingress direction.
Step 10	(Optional) Repeat Step 9 to configure all SPAN sources.	
Step 11	filter vlan {<i>number</i> <i>range</i>} Example: <pre>switch(config-monitor)# filter vlan 3-5, 7</pre>	Configures which VLANs to select from the configured sources. You can configure one or more VLANs, as either a series of comma-separated entries or a range of numbers Note A FEX port that is configured as a SPAN source does not support VLAN filters. Note Filters are not supported when the source is either FC interface or VSAN.
Step 12	(Optional) Repeat Step 11 to configure all source VLANs to filter.	
Step 13	(Optional) filter access-group <i>acl-filter</i> Example: <pre>switch(config-monitor)# filter access-group ACL1</pre>	Associates an ACL with the SPAN session. Note Filters are not supported when the source is either FC interface or VSAN.
Step 14	Required: destination interface <i>type slot/port</i> Example: <pre>switch(config-monitor)# destination interface ethernet 2/5</pre>	Configures a destination for copied source packets. Note FC ports are not supported as a destination interface. Note The SPAN destination port must be either an access port or a trunk port. Note You must enable monitor mode on the destination port. You can configure the CPU as the SPAN destination for the following platform switches:

	Command or Action	Purpose
		• Cisco Nexus 9200 Series switches (beginning with Cisco NX-OS Release 7.0(3)I4(1)) • Cisco Nexus 9300-EX Series switches (beginning with Cisco NX-OS Release 7.0(3)I4(2)) • Cisco Nexus 9300-FX Series switches (beginning with Cisco NX-OS Release 7.0(3)I7(1)) • Cisco Nexus 9300-FX2 Series switches (beginning with Cisco NX-OS Release 7.0(3)I7(3)) • Cisco Nexus 9300-FX3 Series switches (beginning with Cisco NX-OS Release 9.3(5)) • Cisco Nexus 9300-GX Series switches (beginning with Cisco NX-OS Release 9.3(3)) • Cisco Nexus 9500-EX Series switches with -EX/-FX line cards To do so, enter sup-eth 0 for the interface type.
Step 15	Required: no shut Example: <pre>switch(config-monitor)# no shut</pre>	Enables the SPAN session. By default, the session is created in the shut state.
Step 16	(Optional) show monitor session {all session-number range session-range} [brief] Example: <pre>switch(config-monitor)# show monitor session 3</pre>	Displays the SPAN configuration.
Step 17	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring UDF-Based SPAN

You can configure the device to match on user-defined fields (UDFs) of the outer or inner packet fields (header or payload) and to send the matching packets to the SPAN destination. Doing so can help you to analyze and isolate packet drops in the network.

Before you begin

Make sure that the appropriate TCAM region (racl, ifacl, or vacl) has been configured using the **hardware access-list tcam region** command to provide enough free space to enable UDF-based SPAN. For more information, see the "Configuring ACL TCAM Region Sizes" section in the *Cisco Nexus 9000 Series NX-OS Security Configuration Guide*.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	udf udf-name offset-base offset length Example: <pre>switch(config)# udf udf-x packet-start 12 1 switch(config)# udf udf-y header outer 13 20 2</pre>	Defines the UDF as follows: • udf-name—Specifies the name of the UDF. You can enter up to 16 alphanumeric characters for the name. • offset-base—Specifies the UDF offset base as follows, where header is the packet header to consider for the offset: packet-start header {outer inner {I3 I4}}. • offset—Specifies the number of bytes offset from the offset base. To match the first byte from the offset base (Layer 3/Layer 4 header), configure the offset as 0. • length—Specifies the number of bytes from the offset. Only 1 or 2 bytes are supported. To match additional bytes, you must define multiple UDFs. You can define multiple UDFs, but Cisco recommends defining only required UDFs.
Step 3	hardware access-list tcam region {racl ifacl vacl } qualify qualifier-name Example: <pre>switch(config)# hardware access-list tcam region racl qualify ing-l3-span-filter</pre>	Attaches the UDFs to one of the following TCAM regions: • racl—Applies to Layer 3 ports. • ifacl—Applies to Layer 2 ports • vacl—Applies to source VLANs. You can attach up to 8 UDFs to a TCAM region. Note

	Command or Action	Purpose
		When the UDF qualifier is added, the TCAM region goes from single wide to double wide. Make sure enough free space is available; otherwise, this command will be rejected. If necessary, you can reduce the TCAM space from unused regions and then re-enter this command. For more information, see the "Configuring ACL TCAM Region Sizes" section in the Cisco Nexus 9000 Series NX-OS Security Configuration Guide. Note The no form of this command detaches the UDFs from the TCAM region and returns the region to single wide.
Step 4	Required: copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 5	Required: reload Example: <pre>switch(config)# reload</pre>	Reloads the device. Note Your UDF configuration is effective only after you enter copy running-config startup-config + reload .
Step 6	ip access-list span-acl Example: <pre>switch(config)# ip access-list span-acl-udf-only switch(config-acl)#</pre>	Creates an IPv4 access control list (ACL) and enters IP access list configuration mode.
Step 7	Enter one of the following commands: • permit udf udf-name value mask • permit ip source destination udf udf-name value mask Example: <pre>switch(config-acl)# permit udf udf-x 0x40 0xF0 udf-y 0x1001 0xF00F</pre> Example: <pre>switch(config-acl)# permit ip 10.0.0./24 any udf udf-x 0x02 0x0F udf-y 0x1001 0xF00F</pre>	Configures the ACL to match only on UDFs (example 1) or to match on UDFs along with the current access control entries (ACEs) for the outer packet fields (example 2). A single ACL can have ACEs with and without UDFs together. Each ACE can have different UDF fields to match, or all ACEs can match for the same list of UDFs.
Step 8	(Optional) copy running-config startup-config Example:	Copies the running configuration to the startup configuration.

	Command or Action	Purpose
	<code>switch(config)# copy running-config startup-config</code>	

Configuring SPAN Truncation

You can configure truncation for local and SPAN source sessions only.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	monitor session <i>session number</i> Example: <pre>switch(config)# monitor session 5 switch(config-monitor)#</pre>	Enters monitor configuration mode for the specified SPAN session.
Step 3	source interface <i>type slot/port [rx tx both]</i> Example: <pre>switch(config-monitor)# source interface ethernet 1/5 both</pre>	Configures the source interface.
Step 4	mtu <i>size</i> Example: <pre>switch(config-monitor)# mtu 320</pre> Example: <pre>switch(config-monitor)# mtu ? <320-1518> Enter the value of MTU truncation size for SPAN packets</pre>	Configures the MTU size for truncation. Any SPAN packet that is larger than the configured MTU size is truncated to the configured size. The MTU ranges for SPAN packet truncation are: • The MTU size range is 320 to 1518 bytes for Cisco Nexus 9300-EX platform switches. • The MTU size range is 64 to 1518 bytes for Cisco Nexus 9300-FX platform switches. • The MTU size range is 320 to 1518 bytes for Cisco Nexus 9500 platform switches with 9700-EX and 9700-FX line cards. • The MTU size is 343 bytes (excluding FCS) for Cisco Nexus 9808 platform switches.

	Command or Action	Purpose
Step 5	destination interface <i>type slot/port</i> Example: <pre>switch(config-monitor)# destination interface Ethernet 1/39</pre>	Configures the Ethernet SPAN destination port.
Step 6	no shut Example: <pre>switch(config-monitor)# no shut</pre>	Enables the SPAN session. By default, the session is created in the shut state.
Step 7	(Optional) show monitor session session Example: <pre>switch(config-monitor)# show monitor session 5</pre>	Displays the SPAN configuration.
Step 8	copy running-config startup-config Example: <pre>switch(config-monitor)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring SPAN for Multicast Tx Traffic Across Different LSE Slices

Beginning with Cisco NX-OS Release 7.0(3)I7(1), you can configure SPAN for multicast Tx traffic across different leaf spine engine (LSE) slices on Cisco Nexus 9300-EX platform switches.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] hardware multicast global-tx-span Example: <pre>switch(config)# hardware multicast global-tx-span</pre>	Configures SPAN for multicast Tx traffic across different leaf spine engine (LSE) slices. Note Beginning from Cisco NX-OS Release 10.2(2)F, if source and destination are on different slices, use this command for multicast SPAN Tx.
Step 3	copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

	Command or Action	Purpose
Step 4	reload Example: <code>switch(config)# reload</code>	Reloads the device.

Configuring SPAN to CPU

Introduction

A SPAN-to-CPU is for troubleshooting packet flow through Cisco Nexus 9000 Series switches. Similarly, to a normal SPAN or Encapsulated Remote SPAN (ERSPAN) session, a SPAN-to-CPU monitor session involves the definition of one or more source interfaces and traffic directions. Any traffic that matches the direction (TX, RX, or both) defined on a source interface is replicated to the supervisor CPU. This traffic is filtered and analyzed with the use of ethanalyzer or saved to a local storage device for reviewing the results.

To verify whether packets generated by the CPU of a Cisco Nexus 9000 Series Switches are transmitted out of a specific interface, Cisco recommends using a packet capture utility on the remote device connected to the interface.

1. Configuring SPAN as CPU destination

You must be able to configure CPU as monitor session destination and same must be configured on hardware. On Tahoe platforms, this configuration is supported for local span only as there is no customer requirement to support it for ERSpan termination session. The same will be supported for N9K-C9508-FM-R2.

2. Analyzing SPAN Traffic

When SPAN traffic reaches mentioned supervisor CPU. The modules identify as SPAN packets and takes necessary actions and ethanalyzer displays these packets. The Ethanalyzer control plane packet capture utility can be used to view traffic replicated to the CPU. The mirror keyword in the Ethanalyzer command filters traffic such that only traffic replicated by a SPAN-to-CPU monitor session is shown. Ethanalyzer capture and display filters can be used to further limit the traffic displayed.

3. Limiting SPAN traffic rate

Spanned traffic for CPU must be rate limited to avoid control plane disruption. Ethanalyzer uses libpcap module for processing, stripping, and decoding packet headers. Ethanalyzer uses mirror option to display the span traffic reaching supervisor CPU. To match SPAN to CPU a separate span class is created. All the traffic will be created as SPAN class and separate rate is created for this class as Control Plane Policing (COPP). The COPP traffic rate limit will be 50 kbps.

4. Filtering ACL

This will give customers the ability to choose the traffic which they want to monitor. This feature will be supported on all kind of monitor session. For span to cpu this particularly important as traffic will be rate limited and so, it becomes important to categorize the traffic which is intended to be spanned.

Guidelines and Limitations

SPAN-to-CPU has the following configuration guidelines and limitations:

- No ACL Filtering is supported on inband sources.

- Sources such as Physical Interfaces (L2 and L3), port channels, and L3 subinterface are supported with ACL filter.
- ACL Filter is supported for Rx sources only.
- No ACL filtering supported on VLAN sources.
- Configuring multiple span sessions for the same source is not supported.
- MTU truncation is not supported on N9K-X9636C-R, N9K-X9636Q-R, N9K-X9636C-RX, N9K-X96136YC-R, N9K-X9624D-R2, N9K-C9508-FM-R, N9K-C9504-FM-R, N9K-C9508-FM-R2, N9K-C9504-FM-R2, N3K-C36180YC-R, N3K-C3636C-R, and N3K-C36480LD-R2.
- ACL filters are not supported on N9K-X9624D-R2 Line card until Cisco NX-OS release 10.2(2)F.
- Beginning with Cisco NX-OS Release 10.2(3)F, ACL filters is supported on N9K-X9624D-R2 Line card.

Configuring SPAN to CPU

You can configure SPAN to CPU.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	configure CPU as SPAN Example: switch(config-monitor)# destination interface sup-eth0	Configures the CPU as the SPAN destination.
Step 3	configure ACL Filter Example: switch(config-monitor)# filter access-group <acl_filter_name>	Configures the access list which will be honored for filtering.
Step 4	configure ethanalyzer Example: switch# ethanalyzer local interface inband mirror	Displays spanned packets.

Example

This example shows the output of monitor session.

```
show monitor session 1 session 1
type : local
```

```

state : up
acl-name : acl-name not specified
source intf :
rx : Eth3/44
tx : Eth3/44
both : Eth3/44
source VLANs :
rx :
tx :
both :
filter VLANs : filter not specified
source fwd drops :
destination ports : sup-eth0
PFC On Interfaces :
source VSANs :
rx :

```

This example shows the output of copp.

```

# show policy-map interface control-plane | begin span
class-map copp-system-p-class-span (match-any)
match exception span
set cos 0
police cir 50 pps , bc 256 packets
module 1 : <Designated Module>
conformed 910228778 bytes;
7217965 packets;
violated 7217965 bytes;
0 packets;
module 3 :
conformed 0 bytes;
0 packets;
violated 0 bytes;
0 packets;
0 packets;

```

Shutting Down or Resuming a SPAN Session

You can shut down SPAN sessions to discontinue the copying of packets from sources to destinations. You can shut down one session in order to free hardware resources to enable another session. By default, SPAN sessions are created in the shut state.

You can resume (enable) SPAN sessions to resume the copying of packets from sources to destinations. In order to enable a SPAN session that is already enabled but operationally down, you must first shut it down and then enable it.

You can configure the shut and enabled SPAN session states with either a global or monitor configuration mode command.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	[no] monitor session {session-range all} shut Example: <pre>switch(config)# monitor session 3 shut</pre>	Shuts down the specified SPAN sessions. By default, sessions are created in the shut state. The no form of the command resumes (enables) the specified SPAN sessions. By default, sessions are created in the shut state. Note If a monitor session is enabled but its operational status is down, to enable the session, you must first specify the monitor session shut command followed by the no monitor session shut command.
Step 3	monitor session session-number Example: <pre>switch(config)# monitor session 3 switch(config-monitor)#</pre>	Enters the monitor configuration mode. The new session configuration is added to the existing session configuration.
Step 4	[no] shut Example: <pre>switch(config-monitor)# shut</pre>	Shuts down the SPAN session. By default, the session is created in the shut state. The no form of the command enables the SPAN session. By default, the session is created in the shut state.
Step 5	(Optional) show monitor Example: <pre>switch(config-monitor)# show monitor</pre>	Displays the status of SPAN sessions.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Verifying the SPAN Configuration

To display the SPAN configuration, perform one of the following tasks:

Command	Purpose
show monitor session {all session-number range session-range} [brief]	Displays the SPAN session configuration.
show monitor session [session-id all] stats	Displays the SPAN session stats for Cisco Nexus 9808 platform switches.
clear monitor session [session-id all] stats [both rx tx]	Clears the SPAN session stats for Cisco Nexus 9808 platform switches.

Configuration Examples for SPAN

Configuration Example for a SPAN Session

To configure a SPAN session, follow these steps:

Procedure

- Step 1** Configure destination ports in access mode and enable SPAN monitoring.

Example:

```
switch# configure terminal
switch(config)# interface ethernet 2/5
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# no shut
switch(config-if)# exit
switch(config)#
```

- Step 2** Configure a SPAN session.

Example:

```
switch(config)# no monitor session 3
switch(config)# monitor session 3
switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx
switch(config-monitor)# source interface port-channel 2
switch(config-monitor)# source interface sup-eth 0 both
switch(config-monitor)# source vlan 3, 6-8 rx
switch(config-monitor)# source interface ethernet 101/1/1-3
switch(config-monitor)# filter vlan 3-5, 7
switch(config-monitor)# destination interface ethernet 2/5
switch(config-monitor)# no shut
switch(config-monitor)# exit
switch(config)# show monitor session 3
switch(config)# copy running-config startup-config
```

Example:

```
switch(config)# monitor session 1
switch(config-monitor)# source interface fc 1/9/1
switch(config-monitor)# source interface san-port-channel 171
switch(config-monitor)# source vsan 3701
switch(config-monitor)# destination interface ethernet 1/8
switch(config-monitor)# no shutdown
switch(config-monitor)# exit
switch(config)# show monitor session 1
switch(config)# copy running-config startup-config
```


Configuration Example for a Unidirectional SPAN Session

To configure a unidirectional SPAN session, follow these steps:

Procedure

- Step 1** Configure destination ports in access mode and enable SPAN monitoring.

Example:

```
switch# configure terminal
switch(config)# interface ethernet 2/5
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# no shut
switch(config-if)# exit
switch(config)#
```

- Step 2** Configure a SPAN session.

Example:

```
switch(config)# no monitor session 3
switch(config)# monitor session 3 rx
switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx
switch(config-monitor)# filter vlan 3-5, 7
switch(config-monitor)# destination interface ethernet 2/5
switch(config-monitor)# no shut
switch(config-monitor)# exit
switch(config)# show monitor session 3
switch(config)# copy running-config startup-config
```

Configuration Example for a SPAN ACL

This example shows how to configure a SPAN ACL:

```
switch# configure terminal
switch(config)# ip access-list match_11_pkts
switch(config-acl)# permit ip 11.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# ip access-list match_12_pkts
switch(config-acl)# permit ip 12.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# vlan access-map span_filter 5
switch(config-access-map)# match ip address match_11_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# vlan access-map span_filter 10
switch(config-access-map)# match ip address match_12_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# monitor session 1
switch(config-erspan-src)# filter access_group span_filter
```

Configuration Examples for UDF-Based SPAN

This example shows how to configure UDF-based SPAN to match on the inner TCP flags of an encapsulated IP-in-IP packet using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + Outer IP (20) + Inner IP (20) + Inner TCP (20, but TCP flags at 13th byte)
- Offset from packet-start: $14 + 20 + 20 + 13 = 67$
- UDF match value: 0x20
- UDF mask: 0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify ing-l3-span-filter
copy running-config startup-config
reload
ip access-list acl-udf
permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1
source interface Ethernet 1/1
filter access-group acl-udf
```

This example shows how to configure UDF-based SPAN to match regular IP packets with a packet signature (DEADBEEF) at 6 bytes after a Layer 4 header start using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + IP (20) + TCP (20) + Payload: 112233445566DEADBEEF7788
- Offset from Layer 4 header start: $20 + 6 = 26$
- UDF match value: 0xDEADBEEF (split into two-byte chunks and two UDFs)
- UDF mask: 0xFFFFFFFF

```
udf udf_pktsig_msb header outer 14 26 2
udf udf_pktsig_lsb header outer 14 28 2
hardware access-list tcam region racl qualify ing-l3-span-filter
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1
source interface Ethernet 1/1
filter access-group acl-udf-pktsig
```

Configuration Example for SPAN Truncation

This example shows how to configure SPAN truncation for use with MPLS stripping:

```
mpls strip
ip access-list mpls
statistics per-entry
```

```

20 permit ip any any redirect Ethernet1/5
interface Ethernet1/5
switchport
switchport mode trunk
mtu 9216
no shutdown
monitor session 1
source interface Ethernet1/5 tx
mtu 64
destination interface Ethernet1/6
no shut

```

Configuration Examples for Multicast Tx SPAN Across LSE Slices

This example shows how to configure multicast Tx SPAN across LSE slices for Cisco Nexus 9300-EX platform switches. It also shows sample output before and after multicast Tx SPAN is configured.

Before Multicast Tx SPAN Is Configured

```
switch# show interface eth1/15-16, ethernet 1/27 counters
```

```

-----
Port                InOctets      InUcastPkts
-----
Eth1/15              580928        0
Eth1/16               239           0
Eth1/27               0             0

```

```

-----
Port                InMcastPkts   InBcastPkts
-----
Eth1/15              9077          0
Eth1/16               1             0
Eth1/27               0             0

```

```

-----
Port                OutOctets      OutUcastPkts
-----
Eth1/15              453           0
Eth1/16             581317        0
Eth1/27               0             0

```

```

-----
Port                OutMcastPkts   OutBcastPkts
-----
Eth1/15               4             0
Eth1/16             9080          0
Eth1/27               0             0

```

Configuring Multicast Tx SPAN

```

switch(config)# hardware multicast global-tx-span
Warning: Global Tx SPAN setting changed, please save config and reload
switch(config)# copy running-config start-up config
[#####] 100%
Copy complete.
switch(config)# reload
This command will reboot the system. (y/n)? [n] y

```

After Multicast Tx SPAN Is Configured

```
switch# show interface eth1/15-16, eth1/27 counters
```

Port	InOctets	InUcastPkts
Eth1/15	392576	0
Eth1/16	0	0
Eth1/27	0	0
Port	InMcastPkts	InBcastPkts
Eth1/15	6134	0
Eth1/16	0	0
Eth1/27	0	0
Port	OutOctets	OutUcastPkts
Eth1/15	0	0
Eth1/16	392644	0
Eth1/27	417112	0
Port	OutMcastPkts	OutBcastPkts
Eth1/15	0	0
Eth1/16	6135	0
Eth1/27	6134	0

Additional References

Related Documents

Related Topic	Document Title
FEX	<i>Cisco Nexus 2000 Series NX-OS Fabric Extender Software Configuration Guide for Cisco Nexus 9000 Series Switches</i>



CHAPTER 23

Configuring ERSPAN

This chapter describes how to configure an encapsulated remote switched port analyzer (ERSPAN) to transport mirrored traffic in an IP network on Cisco NX-OS devices.

- [About ERSPAN, on page 383](#)
- [Prerequisites for ERSPAN, on page 385](#)
- [Guidelines and Limitations for ERSPAN, on page 385](#)
- [Default Settings, on page 389](#)
- [Configuring ERSPAN, on page 390](#)
- [Verifying the ERSPAN Configuration, on page 403](#)
- [Configuration Examples for ERSPAN, on page 404](#)

About ERSPAN

ERSPAN transports mirrored traffic over an IPv4 or IPv6 network, which provides remote monitoring of multiple switches across your network. The traffic is encapsulated at the source router and is transferred across the network. The packet is decapsulated at the destination router and then sent to the destination interface. Another method is that the destination can be the analyzer itself, which needs to understand the ERSPAN encapsulation format to parse the packet and access the inner (SPAN copy) frame.

ERSPAN Sources

The interfaces from which traffic can be monitored are called ERSPAN sources. Sources designate the traffic to monitor and whether to copy ingress, egress, or both directions of traffic. ERSPAN sources include the following:

- Ethernet ports (but not subinterfaces)
- Port channels
- The inband interface to the control plane CPU



Note When you specify the supervisor inband interface as a SPAN source, the device monitors all packets that are sent by the Supervisor CPU.



Note If you use the supervisor inband interface as a SPAN source, all packets generated by the supervisor hardware (egress) are monitored.

Rx is from the perspective of the ASIC (traffic egresses from the supervisor over the inband and is received by the ASIC/SPAN).

- VLANs
 - When a VLAN is specified as an ERSPAN source, all supported interfaces in the VLAN are ERSPAN sources.
 - VLANs can be ERSPAN sources only in the ingress direction, except for Cisco Nexus 9300-EX/-FX/-FX2/-FX3/-GX series platform switches, and Cisco Nexus 9500 series platform switches with -EX/-FX line cards.



Note A single ERSPAN session can include mixed sources in any combination of the above.

ERSPAN Destination

Destination ports receive the copied traffic from ERSPAN sources. The destination port is a port that is connected to the device such as a Remote Monitoring (RMON) probe or security device that can receive and analyze the copied packets from single or multiple source port. Destination ports do not participate in any spanning tree instance or any Layer 3 protocols

Cisco Nexus 9200, 9300-EX, 9300-FX, and 9300-FX2 platform switches support an ERSPAN destination session configured on physical or port-channel interfaces in switchport mode through the use of GRE header traffic flow. The source IP address should be configured on the default VRF. Multiple ERSPAN destination sessions should be configured with the same source IP address.

ERSPAN Sessions

You can create ERSPAN sessions that designate sources to monitor.

Localized ERSPAN Sessions

An ERSPAN session is localized when all of the source interfaces are on the same line card.



Note An ERSPAN session with a VLAN source is not localized

ERSPAN Truncation

Beginning with Cisco NX-OS Release 7.0(3)I7(1), you can configure the truncation of source packets for each ERSPAN session based on the size of the MTU. Truncation helps to decrease ERSPAN bandwidth by

reducing the size of monitored packets. Any ERSPAN packet that is larger than the configured MTU size is truncated to the given size. For ERSPAN, an additional ERSPAN header is added to the truncated packet from 54 to 166 bytes depending on the ERSPAN header type. For example, if you configure the MTU as 300 bytes, the packets are replicated with an ERSPAN header size from 354 to 466 bytes depending on the ERSPAN header type configuration.

ERSPAN truncation is disabled by default. To use truncation, you must enable it for each ERSPAN session.

Prerequisites for ERSPAN

ERSPAN has the following prerequisites:

- You must first configure the ports on each device to support the desired ERSPAN configuration. For more information, see the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide.

Guidelines and Limitations for ERSPAN



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

ERSPAN has the following configuration guidelines and limitations:

- A maximum of 48 source interfaces are supported per ERSPAN session (Rx and Tx, Rx, or Tx).
- ERSPAN destination handles jumbo frames for MTU differently based on the platform. For the following Cisco Nexus 9300 platform switches and Cisco Nexus 9500 platform switches with supporting line cards, ERSPAN destination drops the jumbo frames:
 - Cisco Nexus 9332PQ
 - Cisco Nexus 9372PX
 - Cisco Nexus 9372PX-E
 - Cisco Nexus 9372TX
 - Cisco Nexus 9372TX-E
 - Cisco Nexus 93120TX
- Cisco Nexus 9500 platform switches with the following line cards:
 - Cisco Nexus 9564PX
 - Cisco Nexus 9464TX
 - Cisco Nexus 9464TX2
 - Cisco Nexus 9564TX
 - Cisco Nexus 9464PX
 - Cisco Nexus 9536PQ

- Cisco Nexus 9636PQ
- Cisco Nexus 9432PQ

For the following Cisco Nexus 9200 platform switches and Cisco Nexus 9500 platform switches with supporting line cards, ERSPAN truncates the packets at port MTU, and issues a TX Output error:

- Cisco Nexus 92160YC-X
- Cisco Nexus 92304QC
- Cisco Nexus 9272Q
- Cisco Nexus 9232C
- Cisco Nexus 9236C
- Cisco Nexus 92300YC
- Cisco Nexus 93108TC-EX
- Cisco Nexus 93180LC-EX
- Cisco Nexus 93180YC-EX
- Cisco Nexus 9500 platform switches with the following line cards:
 - Cisco Nexus 9736C-EX
 - Cisco Nexus 97160YC-EX
 - Cisco Nexus 9732C-EX
 - Cisco Nexus 9732C-EXM
- Using the ACL filter to ERSPAN subinterface traffic on the parent interface is not supported on the Cisco Nexus 9200 platform switches.
- Using the ACL filter to ERSPAN subinterface traffic on the parent interface is not supported on the Cisco Nexus 9300-EX/FX/FX2/FX3/GX platform switches.
- ERSPAN mirroring is not supported for PBR traffic.
- ERSPAN with a Type three header is not supported in Cisco NX-OS Release 9.3(3).
- For ERSPAN session limits, see the *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.
- The number of ERSPAN sessions per line card reduces to two if the same interface is configured as a bidirectional source in more than one session.
- Configuring two SPAN or ERSPAN sessions on the same source interface with only one filter is not supported. If the same source is used in multiple SPAN or ERSPAN sessions either all the sessions must have different filters or no sessions should have filters.
- Beginning with Cisco NX-OS Release 9.3(5), the following ERSPAN features are supported on Cisco Nexus 9300-GX platform switch:
 - ERSPAN Type III Header
 - ERSPAN Destination Support

- Packets with FCS errors are not mirrored in an ERSPAN session.
- TCAM carving is not required for SPAN/ERSPAN on the following line cards:
 - Cisco Nexus 9636C-R
 - Cisco Nexus 9636Q-R
 - Cisco Nexus 9636C-RX
 - Cisco Nexus 96136YC-R
 - Cisco Nexus 9624D-R2



Note All other switches supporting SPAN/ERSPAN must use TCAM carving.

- Statistics are not supported for the filter access group.
- An access-group filter in an ERSPAN session must be configured as vlan-accessmap.
- Control plane packets that are generated by the Supervisor cannot be ERSPAN encapsulated or filtered by an ERSPAN access control list (ACL).
- ERSPAN is not supported for management ports.
- ERSPAN does not support destinations on Layer 3 port-channel subinterfaces.
- VLAN as a source is not supported with ERSPAN configuration on R-series linecards and N3K-C36180YC-R, N3KC36480LD-R2, and N3K-C3636C-R platform switches.
- A VLAN can be part of only one session when it is used as an ERSPAN source or filter.
- VLAN ERSPAN monitors only the traffic that leaves or enters Layer 2 ports in the VLAN.
- If you enable ERSPAN on a vPC and ERSPAN packets must be routed to the destination through the vPC, packets that come through the vPC peer link cannot be captured.
- ERSPAN is not supported over a VXLAN overlay.
- ERSPAN copies for multicast packets are made before rewrite. Therefore, the TTL, VLAN ID, any remarking due to egress policy, and so on, are not captured in the ERSPAN copy.
- The timestamp granularity of ERSPAN Type III sessions is not configurable through the CLI. It is 100 picoseconds and driven through PTP.
- ERSPAN works on default and nondefault VRFs, but ERSPAN marker packets work only on the default VRF.
- The same source can be part of multiple sessions.

The following guidelines and limitations apply to egress (Tx) ERSPAN:

- The flows for post-routed unknown unicast flooded packets are in the ERSPAN session, even if the ERSPAN session is configured to not monitor the ports on which this flow is forwarded. This limitation applies to Network Forwarding Engine (NFE) and NFE2-enabled EOR switches and ERSPAN sessions that have TX port sources.

- For ERSPAN Tx multicast for Layer 2, ERSPAN copies are created independent of multicast replication. Due to this, multicast and SPAN packet have different values for VLAN tag, which is the ingress interface VLAN ID.
- The following guidelines and limitations apply to ingress (Rx) ERSPAN:
 - VLAN sources are spanned only in the Rx direction.
 - Session filtering functionality (VLAN or ACL filters) is supported only for Rx sources.
 - VLANs are supported as ERSPAN sources only in the ingress direction.
- Priority flow control (PFC) ERSPAN has the following guidelines and limitations:
 - It cannot coexist with filters.
 - It is supported only in the Rx direction on physical or port-channel interfaces. It is not supported in the Rx direction on VLAN interfaces or in the Tx direction.
- The following guidelines and limitations apply to FEX ports:
 - If the sources used in bidirectional ERSPAN sessions are from the same FEX, the hardware resources are limited to two ERSPAN sessions.
 - FEX ports are supported as ERSPAN sources in the ingress direction for all traffic and in the egress direction only for known Layer 2 unicast traffic.
 - Cisco Nexus 9300 platform switches do not support ERSPAN destination being connected on a FEX interface. The ERSPAN destination must be connected to a front panel port.
 - VLAN and ACL filters are not supported for FEX ports. It cannot coexist with filters.
- The following guidelines and limitations apply to ERSPAN destination:
 - Cisco Nexus 9200, 9300-EX, 9300-FX, and 9300-FX2 platform switches support an ERSPAN destination session that is configured on physical or port-channel interfaces in switchport mode by using GRE header traffic flow.
 - ERSPAN destination cannot coexist with other tunnel features such as MPLS and VXLAN for Cisco Nexus 9200, 9300, 9300-EX, 9300-FX, and 9300-FX2 platform switches.
 - On Cisco Nexus 9300-GX switches, dot1q-tagged broadcast or multicast packets passing through a device where the ERSPAN destination session is active, get tagged with the native VLAN instead of the correct VLAN due to a hardware limitation.
 - ERSPAN destination supports only default VRF.
 - Cisco Nexus 9300-EX/FX switches cannot serve as an ERSPAN destination for Cisco Nexus 3000 and non-EX/FX Cisco Nexus 9000 switches.
- Beginning with Cisco NX-OS Release 10.1(2), ERSPAN is supported on the Cisco N9K-X9624D-R2 Line Card.
- The following guidelines and limitations apply to ERSPAN over IPv6:
 - Beginning with Cisco NX-OS Release 10.2(1)F, the ERSPAN over IPv6 feature is supported on Cisco Nexus 9300-GX2, 9300-GX, 9300-FXP, 9300-FX2, 9300-EX, 9300-FX3, 9300-FX3S, and

9300-FX3P platform switches and N9K-X9716D-GX, N9K-X9736C-EX, N9K-X9732C-EX(X86_64 Atom), N9K-X9732C-EXM, N9K-X97160YC-EX, and N9K-X9736C-FX line cards.

- This feature is not supported for Load balancing across egress port-channel members and egress ECMP path.
- This feature is not supported for header-type 3, udf in filter ACL, and marker-packets.
- This feature is not supported for FEX host interface as ERSPAN source with IPv6.
- Beginning with Cisco NX-OS Release 10.2(3)F, IPv6 is supported on ERSPAN destination/termination on Cisco Nexus 9300-GX2, 9300-GX, 9300-FXP, 9300-FX2, 9300-EX, 9300-FX3, 9300-FX3S, and 9300-FX3P platform switches and N9K-X9716D-GX, N9K-X9736C-EX, N9K-X9732C-EX(X86_64 Atom), N9K-X9732C-EXM, N9K-X97160YC-EX, and N9K-X9736C-FX line cards.
- The following guidelines and limitations are applicable:
 - Only VRF default is supported.
 - You can only have one IPv6 address per switch.
 - This feature is not supported with other tunnel features.
 - You can bring up four ERSPAN destination sessions at a time.
 - ERSPAN ID is unique per session and the range is 1–32.
- Beginning with Cisco NX-OS Release 10.3(1)F, ERSPAN is supported on Cisco Nexus 9808 platform switches.
 - Only RX is supported on ERSPAN.
 - Type 3 header is not supported.
 - ERSPAN destination/termination is not supported.
- When configuring local SPAN sessions or ERSPAN-source monitor sessions with a filter access-group rule, we recommend to configure the necessary sub-commands of the VLAN access-map too, such as the match rule.

Default Settings

The following table lists the default settings for ERSPAN parameters.

Table 21: Default ERSPAN Parameters

Parameters	Default
ERSPAN sessions	Created in the shut state

Configuring ERSPAN



Note Be aware that the Cisco NX-OS commands for this feature may differ from those commands used in Cisco IOS.

Configuring an ERSPAN Source Session

You can configure an ERSPAN session on the local device only. By default, ERSPAN sessions are created in the shut state.



Note ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	monitor erspan origin ip-address ip-address global or monitor erspan origin ipv6-address ipv6-address global Example: <pre>switch(config)# monitor erspan origin ip-address 10.0.0.1 global switch(config)# monitor erspan origin ipv6-address 2001:DB8:1::1 global</pre>	Configures the ERSPAN global origin IPv4 or IPv6 address.
Step 3	no monitor session {session-number all} Example: <pre>switch(config)# no monitor session 3</pre>	Clears the configuration of the specified ERSPAN session. The new session configuration is added to the existing session configuration.
Step 4	monitor session {session-number all} type erspan-source [shut] Example: <pre>switch(config)# monitor session 3 type erspan-source switch(config-erspan-src)#</pre>	Configures an ERSPAN Type II source session. By default the session is bidirectional. The optional keyword shut specifies a shut state for the selected session.

	Command or Action	Purpose
Step 5	description <i>description</i> Example: <pre>switch(config-erspan-src) # description erspan_src_session_3</pre>	Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters.
Step 6	source { interface type [tx rx both] vlan { number range } [rx]} Example: <pre>switch(config-erspan-src) # source interface ethernet 2/1-3, ethernet 3/1 rx</pre> Example: <pre>switch(config-erspan-src) # source interface port-channel 2</pre> Example: <pre>switch(config-erspan-src) # source interface sup-eth 0 rx</pre> Example: <pre>switch(config-erspan-src) # source vlan 3, 6-8 rx</pre> Example: <pre>switch(config-erspan-src) # source interface ethernet 101/1/1-3</pre>	Configures the sources and traffic direction in which to copy packets. You can enter a range of Ethernet ports, a port channel, an inband interface, a range of VLANs, or a satellite port or host interface port channel on the Cisco Nexus 2000 Series Fabric Extender (FEX). You can configure one or more sources, as either a series of comma-separated entries or a range of numbers. You can specify the traffic direction to copy as ingress, egress, or both. For a unidirectional session, the direction of the source must match the direction specified in the session. Note Source VLANs are supported only in the ingress direction. Source FEX ports are supported in the ingress direction for all traffic and in the egress direction only for known Layer 2 unicast traffic. Supervisor as a source is only supported in the Rx direction.
Step 7	(Optional) Repeat Step 7 to configure all ERSPAN sources.	—
Step 8	filter vlan { <i>number</i> <i>range</i> } Example: <pre>switch(config-erspan-src) # filter vlan 3-5, 7</pre>	Configures which VLANs to select from the configured sources. You can configure one or more VLANs, as either a series of comma-separated entries or a range of numbers. For information on the VLAN range, see the Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide. Note A FEX port that is configured as an ERSPAN source does not support VLAN filters.
Step 9	(Optional) Repeat Step 9 to configure all source VLANs — to filter.	—
Step 10	(Optional) filter access-group <i>acl-filter</i> Example:	Associates an ACL with the ERSPAN session. (You can create an ACL using the standard ACL configuration process. For more

	Command or Action	Purpose
	<pre>switch(config-erspan-src)# filter access-group ACL1</pre>	information, see the <i>Cisco Nexus 9000 Series NX-OS Security Configuration Guide</i>.) Note Before executing this command, configure ip access list and associated vlan access map. See Configuring ERSPAN ACL.
Step 11	destination ip <i>ip-address</i> Example: <pre>switch(config-erspan-src)# destination ip 10.1.1.1 switch(config-erspan-src)# destination ipv6 2001:DB8:1::1</pre>	destination ipv6 <i>ipv6-address</i> Configures the destination IPv4 or IPv6 address in the ERSPAN session. Note Only one destination IPv4 or IPv6 address is supported per ERSPAN source session.
Step 12	erspan-id <i>erspan-id</i> Example: <pre>switch(config-erspan-src)# erspan-id 5</pre>	Configures the ERSPAN ID for the ERSPAN source session. The ERSPAN range is from 1 to 1023.
Step 13	vrf <i>vrf-name</i> Example: <pre>switch(config-erspan-src)# vrf default</pre>	Configures the virtual routing and forwarding (VRF) instance that the ERSPAN source session uses for traffic forwarding. The VRF name can be any case-sensitive, alphanumeric string up to 32 characters.
Step 14	(Optional) ip ttl <i>ttl-number</i> Example: <pre>switch(config-erspan-src)# ip ttl 25</pre>	Configures the IP time-to-live (TTL) value for the ERSPAN traffic. The range is from 1 to 255.
Step 15	(Optional) ip dscp <i>dscp-number</i> Example: <pre>switch(config-erspan-src)# ip dscp 42</pre>	Configures the differentiated services code point (DSCP) value of the packets in the ERSPAN traffic. The range is from 0 to 63.
Step 16	no shut Example: <pre>switch(config-erspan-src)# no shut</pre>	Enables the ERSPAN source session. By default, the session is created in the shut state.
Step 17	exit Example: <pre>switch(config-erspan-src)# exit switch(config)#</pre>	Exits the monitor configuration mode.
Step 18	(Optional) show monitor session {all <i>session-number</i> range <i>session-range</i>} [brief] Example: <pre>switch(config)# show monitor session 3</pre>	Displays the ERSPAN session configuration.

	Command or Action	Purpose
Step 19	(Optional) show running-config monitor Example: <code>switch(config)# show running-config monitor</code>	Displays the running ERSPAN configuration.
Step 20	(Optional) show startup-config monitor Example: <code>switch(config)# show startup-config monitor</code>	Displays the ERSPAN startup configuration.
Step 21	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Shutting Down or Activating an ERSPAN Session

You can shut down ERSPAN sessions to discontinue the copying of packets from sources to destinations. You can shut down one session in order to free hardware resources to enable another session. By default, ERSPAN sessions are created in the shut state.

You can enable ERSPAN sessions to activate the copying of packets from sources to destinations. To enable an ERSPAN session that is already enabled but operationally down, you must first shut it down and then enable it. You can shut down and enable the ERSPAN session states with either a global or monitor configuration mode command.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	monitor session {session-range all} shut Example: <code>switch(config)# monitor session 3 shut</code>	Shuts down the specified ERSPAN sessions. By default, sessions are created in the shut state.
Step 3	no monitor session {session-range all} shut Example: <code>switch(config)# no monitor session 3 shut</code>	Resumes (enables) the specified ERSPAN sessions. By default, sessions are created in the shut state. If a monitor session is enabled but its operational status is down, then to enable the session, you must first specify the monitor

	Command or Action	Purpose
		session shut command followed by the no monitor session shut command.
Step 4	monitor session session-number type erspan-source Example: <pre>switch(config)# monitor session 3 type erspan-source switch(config-erspan-src)#</pre>	Enters the monitor configuration mode for the ERSPAN source type. The new session configuration is added to the existing session configuration.
Step 5	shut Example: <pre>switch(config-erspan-src)# shut</pre>	Shuts down the ERSPAN session. By default, the session is created in the shut state.
Step 6	no shut Example: <pre>switch(config-erspan-src)# no shut</pre>	Enables the ERSPAN session. By default, the session is created in the shut state.
Step 7	exit Example: <pre>switch(config-erspan-src)# exit switch(config)#</pre>	Exits the monitor configuration mode.
Step 8	(Optional) show monitor session all Example: <pre>switch(config)# show monitor session all</pre>	Displays the status of ERSPAN sessions.
Step 9	(Optional) show running-config monitor Example: <pre>switch(config)# show running-config monitor</pre>	Displays the ERSPAN running configuration.
Step 10	(Optional) show startup-config monitor Example: <pre>switch(config)# show startup-config monitor</pre>	Displays the ERSPAN startup configuration.
Step 11	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring an ERSPAN ACL

You can create an IPv4 ERSPAN ACL on the device and add rules to it.

Before you begin

To modify the DSCP value or the GRE protocol, you need to allocate a new destination monitor session. A maximum of four destination monitor sessions are supported.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	ip access-list <i>acl-name</i> Example: <pre>switch(config)# ip access-list erspan-acl switch(config-acl)#</pre>	Creates the ERSPAN ACL and enters IP ACL configuration mode. The <i>acl-name</i> argument can be up to 64 characters.
Step 3	[<i>sequence-number</i>] {permit deny} <i>protocol</i> <i>source destination</i> [set-erspan-dscp <i>dscp-value</i>] [set-erspan-gre-PROTO <i>protocol-value</i>] Example: <pre>switch(config-acl)# permit ip 192.168.2.0/24 any set-erspan-dscp 40 set-erspan-gre-PROTO 5555</pre> Example: <pre>switch(config)# ip access-list match_11_pkts switch(config-acl)# permit ip 10.0.0.0/24 any switch(config-acl)# exit</pre>	Creates a rule in the ERSPAN ACL. You can create many rules. The <i>sequence-number</i> argument can be a whole number between 1 and 4294967295. The permit and deny commands support many ways of identifying traffic. The set-erspan-dscp option sets the DSCP value in the ERSPAN outer IP header. The range for the DSCP value is from 0 to 63. The DSCP value configured in the ERSPAN ACL overrides the value configured in the monitor session. If you do not include this option in the ERSPAN ACL, 0 or the DSCP value configured in the monitor session will be set. The set-erspan-gre-PROTO option sets the protocol value in the ERSPAN GRE header. The range for the protocol value is from 0 to 65535. If you do not include this option in the ERSPAN ACL, the default value of 0x88be will be set as the protocol in the GRE header for ERSPAN-encapsulated packets. Each access control entry (ACE) with the set-erspan-gre-PROTO or set-erspan-dscp action consumes one destination monitor session. A maximum of three ACEs with one of these actions is supported per ERSPAN ACL. For example, you can configure one of the following:

	Command or Action	Purpose
		• One ERSPAN session with an ACL having a maximum of three ACEs with the set-erspan-gre-proto or set-erspan-dscp action • One ERSPAN session with an ACL having two ACEs with the set-erspan-gre-proto or set-erspan-dscp action and one additional local or ERSPAN session • A maximum of two ERSPAN sessions with an ACL having one ACE with the set-erspan-gre-proto or set-erspan-dscp action
Step 4	vlan access-map erspan-acl map name [sequence-number] Example: <pre>switch(config)# vlan access-map erspan_filter</pre>	Enters VLAN access-map configuration mode for the VLAN access map specified. If the VLAN access map does not exist, the device creates it. If you do not specify a sequence number, the device creates a new entry whose sequence number is 10 greater than the last sequence number in the access map.
Step 5	match ip address acl-name Example: <pre>switch(config-access-map)# match ip address erspan-acl</pre>	Specifies an ACL for the access-map entry.
Step 6	action forward Example: <pre>switch(config-access-map)# action forward</pre>	Specifies the action that the device applies to traffic that matches the ACL.
Step 7	exit Example: <pre>switch(config-access-map)# exit</pre>	Exits VLAN access-map configuration mode.
Step 8	monitor session [session-number all] type erspan-source [shut] Example: <pre>switch(config)# monitor session 1 type erspan-source</pre>	Configures an ERSPAN Type II source session. By default, the session is bidirectional. The optional keyword shut specifies a shut state for the selected session.
Step 9	filter access_group name Example:	Associates an ACL with the ERSPAN session. (You can create an ACL using the standard ACL configuration process. For more

	Command or Action	Purpose
	<code>switch(config-erspan-src)# filter access_group erspan_filter</code>	information, see <i>Cisco Nexus 9000 Series NX-OS Security Configuration Guide</i> .)
Step 10	(Optional) copy running-config startup-config Example: <code>switch(config-acl)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Verifying ERSPAN ACL Configuration

To display the ERSPAN ACL configuration, execute the appropriate show commands from the following table.

Command	Purpose
show ip access-lists <i>name</i> Example: <code>switch(config-acl)# show ip access-lists erspan-acl</code>	Displays the ERSPAN ACL configuration.
show vlan access-map <i>name</i> Example: <code>switch(config-acl)# show vlan access-map erspan_filter</code>	Displays information about VLAN access maps.
show monitor session {all session-number range session-range} [brief] Example: <code>switch(config-acl)# show monitor session 1</code>	Displays the ERSPAN session configuration.

Configuring UDF-Based ERSPAN

You can configure the device to match on user-defined fields (UDFs) of the outer or inner packet fields (header or payload) and to send the matching packets to the ERSPAN destination. Doing so can help you to analyze and isolate packet drops in the network.

Before you begin

Make sure that the appropriate TCAM region (racl, ifacl, or vacl) has been configured using the **hardware access-list tcam region** command to provide enough free space to enable UDF-based ERSPAN. For information, see the "Configuring ACL TCAM Region Sizes" section in the Cisco Nexus 9000 Series NX-OS Security Configuration Guide.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	udf udf-name offset-base offset length Example: <pre>switch(config)# udf udf-x packet-start 12 1 switch(config)# udf udf-y header outer 13 20 2</pre>	Defines the UDF as follows: • <i>udf-name</i>—Specifies the name of the UDF. You can enter up to 16 alphanumeric characters for the name. • <i>offset-base</i>—Specifies the UDF offset base as follows, where header is the packet header to consider for the offset: packet-start header {outer inner {13 14}}. • <i>offset</i>—Specifies the number of bytes offset from the offset base. To match the first byte from the offset base (Layer 3/Layer 4 header), configure the offset as 0. • <i>length</i>—Specifies the number of bytes from the offset. Only 1 or 2 bytes are supported. To match additional bytes, you must define multiple UDFs. You can define multiple UDFs, but Cisco recommends defining only required UDFs.
Step 3	hardware access-list tcam region {racl ifacl vACL } qualify udf udf-names Example: <pre>switch(config)# hardware access-list tcam region racl qualify udf udf-x udf-y</pre>	Attaches the UDFs to one of the following TCAM regions: • <i>racl</i>—Applies to Layer 3 ports.—Applies to layer 2 and Layer 3 ports. • <i>ifacl</i>—Applies to Layer 2 ports. • <i>vACL</i>—Applies to source VLANs. You can attach up to 8 UDFs to a TCAM region. Note When the UDF qualifier is added, the TCAM region goes from single wide to double wide. Make sure enough free space is available; otherwise, this command will be rejected. If necessary, you can reduce the TCAM space

	Command or Action	Purpose
		from unused regions and then re-enter this command. For more information, see the "Configuring ACL TCAM Region Sizes" section in the Cisco Nexus 9000 Series NX-OS Security Configuration Guide. Note The no form of this command detaches the UDFs from the TCAM region and returns the region to single wide.
Step 4	Required: copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 5	Required: reload Example: <pre>switch(config)# reload</pre>	Reloads the device. Note Your UDF configuration is effective only after you enter copy running-config startup-config + reload.
Step 6	ip access-list erspan-acl Example: <pre>switch(config)# ip access-list erspan-acl-udf-only switch(config-acl)#</pre>	Creates an IPv4 access control list (ACL) and enters IP access list configuration mode.
Step 7	Enter one of the following commands: • permit udf udf-name value mask • permit ip source destination udf udf-name value mask Example: <pre>switch(config-acl)# permit udf udf-x 0x40 0xF0 udf-y 0x1001 0xF00F</pre> Example: <pre>switch(config-acl)# permit ip 10.0.0.0/24 any udf udf-x 0x02 0x0F udf-y 0x1001 0xF00F</pre>	Configures the ACL to match only on UDFs (example 1) or to match on UDFs along with the current access control entries (ACEs) for the outer packet fields (example 2). A single ACL can have ACEs with and without UDFs together. Each ACE can have different UDF fields to match, or all ACEs can match for the same list of UDFs.
Step 8	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring ERSPAN Truncation

You can configure truncation for local and ERSPAN source sessions only.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	monitor session <i>session-number</i> type erspan-source Example: <pre>switch(config)# monitor session 10 type erspan-source switch(config-erspan-src)#</pre>	Enters monitor configuration mode for the specified ERSPAN session.
Step 3	source interface <i>type slot/port</i> [<i>rx</i> <i>tx</i> <i>both</i>] Example: <pre>switch(config-erspan-src)# source interface ethernet 1/5 both</pre>	Configures the source interface.
Step 4	mtu <i>size</i> Example: <pre>switch(config-erspan-src)# mtu 512</pre> Example: <pre>switch(config-erspan-src)# mtu ? <512-1518> Enter the value of MTU truncation size for ERSPAN packets (erspan header + truncated original packet)</pre>	Configures the MTU size for truncation. Any ERSPAN packet that is larger than the configured MTU size is truncated to the configured size. The MTU ranges for ERSPAN packet truncation are: • The MTU size range is 512 to 1518 bytes for Cisco Nexus 9300-EX Series switches. • The MTU size range is 64 to 1518 bytes for Cisco Nexus 9300-FX Series switches. • The MTU size range is 512 to 1518 bytes for Cisco Nexus 9500 platform switches with 9700-EX and 9700-FX line cards. • The MTU size is 343 bytes (excluding FCS) for Cisco Nexus 9808 platform switches.
Step 5	destination interface <i>type slot/port</i> Example: <pre>switch(config-erspan-src)# destination interface Ethernet 1/39</pre>	Configures the Ethernet ERSPAN destination port.

	Command or Action	Purpose
Step 6	no shut Example: switch(config-erspan-src) # no shut	Enables the ERSPAN session. By default, the session is created in the shut state.
Step 7	(Optional) show monitor session session Example: switch(config-erspan-src) # show monitor session 5	Displays the ERSPAN configuration.
Step 8	copy running-config startup-config Example: switch(config-erspan-src) # copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring an ERSPAN Destination Session

You can configure a ERSPAN destination session to copy packets from a source IP address to destination ports on the local device. By default, ERSPAN destination sessions are created in the shut state.

Before you begin

Ensure that you have already configured the destination ports in switchport monitor mode.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config) #	Enters global configuration mode.
Step 2	interface ethernet slot/port[-port] Example: switch(config) # interface ethernet 2/5 switch(config-if) #	Enters interface configuration mode on the selected slot and port or range of ports.
Step 3	switchport Example: switch(config-if) # switchport	Configures switchport parameters for the selected slot and port or range of ports.
Step 4	switchport mode [access trunk] Example: switch(config-if) # switchport mode trunk	Configures the following switchport modes for the selected slot and port or range of ports: • access • trunk

	Command or Action	Purpose
Step 5	switchport monitor Example: switch(config-if)# switchport monitor	Configures the switchport interface as an ERSPAN destination.
Step 6	Repeat Steps 2 to 5 to configure monitoring on additional ERSPAN destinations.	—
Step 7	no monitor session {session-number all} Example: switch(config-if)# no monitor session 3	Clears the configuration of the specified ERSPAN session. The new session configuration is added to the existing session configuration.
Step 8	monitor session {session-number all} type erspan-destination Example: switch(config-if)# monitor session 3 type erspan-destination switch(config-erspan-dst)#	Configures an ERSPAN destination session.
Step 9	description description Example: switch(config-erspan-dst)# description erspan_dst_session_3	Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters.
Step 10	source ip ip-address Example: switch(config-erspan-dst)# source ip 10.1.1.1 switch(config-erspan-dst)# source ipv6 2001:DB8:1::1	source ipv6 ipv6-address Configures the source IPv4 or IPv6 address in the ERSPAN session. The source IPv4 or IPv6 address is a locally configured IPv4 or IPv6 address. The source IPv4 or IPv6 address in an ERSPAN destination session must match the destination IPv4 or IPv6 address configured in the ERSPAN source session from which the encapsulated data is received. Only one source IPv4 or IPv6 address is supported per ERSPAN destination session. Note IPv6 is supported from Cisco NX-OS Release 10.2(3)F.
Step 11	destination {[interface [type slot/port[-port]] [port-channel channel-number]]} Example: switch(config-erspan-dst)# destination interface ethernet 2/5	Configures a destination for copied source packets. You can configure a destination interface. Note You can configure destination ports as trunk ports.

	Command or Action	Purpose
Step 12	(Optional) Repeat Step 11 to configure all ERSPAN destinations.	—
Step 13	erspan-id <i>erspan-id</i> Example: <code>switch(config-erspan-dst)# erspan-id 5</code>	Configures the ERSPAN ID for the ERSPAN session. The range is from 1 to 1023.
Step 14	no shut Example: <code>switch(config-erspan-dst)# no shut</code>	Enables the ERSPAN destination session. By default, the session is created in the shut state.
Step 15	exit Example: <code>switch(config-erspan-dst)# exit</code>	Exits monitor configuration mode.
Step 16	exit Example: <code>switch(config)# exit</code>	Exits global configuration mode.
Step 17	(Optional) show monitor session { all <i>session-number</i> range <i>session-range</i> } Example: <code>switch(config)# show monitor session 3</code>	Displays the ERSPAN session configuration.
Step 18	(Optional) show running-config monitor Example: <code>switch(config-erspan-src)# show running-config monitor</code>	Displays the running ERSPAN configuration.
Step 19	(Optional) show startup-config monitor Example: <code>switch(config-erspan-src)# show startup-config monitor</code>	Displays the ERSPAN startup configuration.
Step 20	(Optional) copy running-config startup-config Example: <code>switch(config-erspan-src)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Verifying the ERSPAN Configuration

To display the ERSPAN configuration, perform one of the following tasks:

Command	Purpose
show monitor session {all session-number range session-range} [brief]	Displays the ERSPAN session configuration.
show running-config monitor	Displays the running ERSPAN configuration.
show startup-config monitor	Displays the ERSPAN startup configuration.
show monitor session [session-id all] stats	Displays the ERSPAN session stats for Cisco Nexus 9808 platform switches.
clear monitor session [session-id all] stats [both rx tx]	Clears the ERSPAN session stats for Cisco Nexus 9808 platform switches.

Configuration Examples for ERSPAN

Configuration Example for an ERSPAN Source Session Over IPv6

This example shows how to configure an ERSPAN source session over IPv6:

```
switch# configure terminal
switch(config)# monitor erspan origin ipv6-address 2001::10:0:0:9 global
switch(config)# moni session 10 type erspan-source
switch(config-erspan-src)# erspan-id 10
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# source interface ethernet 1/64
switch(config-erspan-src)# destination ip 10.1.1.2
```

Configuration Example for a Unidirectional ERSPAN Session

This example shows how to configure a unidirectional ERSPAN session:

```
switch# configure terminal
switch(config)# interface ethernet 14/30
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# no monitor session 3
switch(config)# monitor session 3 rxswitch(config-erspan-src)# source interface ethernet
2/1-3 rx
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 10.1.1.2
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
switch(config)# show monitor session 1
```

Configuration Example for an ERSPAN ACL

This example shows how to configure an ERSPAN ACL:

```

switch# configure terminal
switch(config)# ip access-list match_10_pkts
switch(config-acl)# permit ip 10.0.0.0/24 any
switch(config-acl)# exit
switch(config)# ip access-list match_172_pkts
switch(config-acl)# permit ip 172.16.0.0/24 any
switch(config-acl)# exit

```

In the case of different ERSPAN destinations where the interesting traffic is chosen based on the defined ACL filters, the last configured session would always have the higher priority.

For example, if Monitor Session 1 is configured; then Monitor Session 2 is configured; then ERSPAN traffic filter works as intended. But, if the user goes back to Monitor Session 1 and re-applies one of the existing configuration line (no new changes in the config); then the spanned traffic switches back to Monitor Session 1.

Configuration Example for a Marker Packet

This example shows how to enable the ERSPAN marker packet with an interval of 2 seconds:

```

switch# configure terminal
switch(config)# monitor erspan origin ip-address 172.28.15.250 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# header-type 3
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 10.1.1.2
switch(config-erspan-src)# source interface ethernet 1/15 both
switch(config-erspan-src)# marker-packet 100
switch(config-erspan-src)# no shut
switch(config-erspan-src)# show monitor session 1
session 1
-----
type                : erspan-source
state               : up
granularity         : nanoseconds
erspan-id           : 1
vrf-name            : default
destination-ip      : 10.1.1.2
ip-ttl              : 16
ip-dscp             : 5
header-type         : 3
origin-ip           : 172.28.15.250 (global)
source intf         :
    rx              : Eth1/15
    tx              : Eth1/15
    both            : Eth1/15
    rx              :
marker-packet       : enabled
packet interval     : 100
packet sent         : 25
packet failed       : 0
egress-intf        :

```

Configuration Examples for UDF-Based ERSPAN

This example shows how to configure UDF-based ERSPAN to match on the inner TCP flags of an encapsulated IP-in-IP packet using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + Outer IP (20) + Inner IP (20) + Inner TCP (20, but TCP flags at 13th byte)
- Offset from packet-start: $14 + 20 + 20 + 13 = 67$
- UDF match value: 0x20
- UDF mask: 0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify udf udf_tcpflags
copy running-config startup-config
reload
ip access-list acl-udf
permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1 type erspan-source
source interface Ethernet 1/1
filter access-group acl-udf
```

This example shows how to configure UDF-based ERSPAN to match regular IP packets with a packet signature (DEADBEEF) at 6 bytes after a Layer 4 header start using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + IP (20) + TCP (20) + Payload: 112233445566DEADBEEF7788
- Offset from Layer 4 header start: $20 + 6 = 26$
- UDF match value: 0xDEADBEEF (split into two-byte chunks and two UDFs)
- UDF mask: 0xFFFFFFFF

```
udf udf_pktsig_msb header outer 13 26 2
udf udf_pktsig_lsb header outer 13 28 2
hardware access-list tcam region racl qualify udf udf_pktsig_msb udf_pktsig_lsb
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1 type erspan-source
source interface Ethernet 1/1
filter access-group acl-udf-pktsig
```

Configuration Example for ERSPAN Truncation

This example shows how to configure ERSPAN truncation for use with MPLS stripping:

```
mpls strip
ip access-list mpls
statistics per-entry
```

```

20 permit ip any any redirect Ethernet1/5

interface Ethernet1/5
  switchport
  switchport mode trunk
  mtu 9216
  no shutdown

monitor session 1
  source interface Ethernet1/5 tx
  mtu 64
  destination interface Ethernet1/6
  no shut
monitor session 21 type erspan-source
  description "ERSPAN Session 21"
  header-type 3
  erspan-id 21
  vrf default
  destination ip 10.1.1.2
  source interface Ethernet1/5 tx
  mtu 64
  no shut
monitor session 22 type erspan-source
  description "ERSPAN Session 22"
  erspan-id 22
  vrf default
  destination ip 10.2.1.2
  source interface Ethernet1/5 tx
  mtu 750
  no shut
monitor session 23 type erspan-source
  description "ERSPAN Session 23"
  header-type 3
  marker-packet 1000
  erspan-id 23
  vrf default
  destination ip 10.3.1.2
  source interface Ethernet1/5 tx
  mtu 1000
  no shut

```

Configuration Example for an ERSPAN Destination Session Over IPv4

This example shows how to configure an ERSPAN destination session over IPv4:

The **destination interface eth1/1** is in switchport monitor mode. This interface can not co-exist with mpls strip, tunnel, nv overlay, vn-segment-vlan-based, mpls segment-routing, mpls evpn, mpls static, mpls oam, mpls l3vpn, mpls ldp, and nv overlay evpn features.

```

switch# monitor session 1 type erspan-destination
switch(config)# erspan-id 1
switch(config-erspan-dst)# source ip 10.1.1.1
switch(config-erspan-dst)# destination interface eth1/1
switch(config-erspan-dst)# no shut
switch(config-erspan-dst)# exit

```

Configuration Example for an ERSPAN Destination Session Over IPv6

This example shows how to configure an ERSPAN destination session over IPv6:

The **destination interface eth1/1** is in switchport monitor mode. This interface can not co-exist with mpls strip, tunnel, nv overlay, vn-segment-vlan-based, mpls segment-routing, mpls evpn, mpls static, mpls oam, mpls l3vpn, mpls ldp, and nv overlay evpn features.

```
switch# monitor session 1 type erspan-destination
switch(config)# erspan-id 1
switch(config-erspan-dst)# source ipv6 2001:DB8:1::1
switch(config-erspan-dst)# destination interface eth1/1
switch(config-erspan-dst)# no shut
switch(config-erspan-dst)# exit
```



CHAPTER 24

Configuring LLDP

This chapter describes how to configure the Link Layer Discovery Protocol (LLDP) in order to discover other devices on the local network.

This chapter contains the following sections:

- [About LLDP, on page 409](#)
- [Guidelines and Limitations for LLDP , on page 411](#)
- [Default Settings for LLDP, on page 412](#)
- [Configuring LLDP, on page 413](#)
- [Verifying the LLDP Configuration, on page 422](#)
- [Configuration Example for LLDP, on page 423](#)

About LLDP

The Cisco Discovery Protocol (CDP) is a device discovery protocol that allows network management applications to automatically discover and learn about other Cisco devices that are connected to the network.

To permit the discovery of non-Cisco devices, the switch also supports the Link Layer Discovery Protocol (LLDP), a vendor-neutral device discovery protocol that is defined in the IEEE 802.1ab standard. LLDP allows 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 is a one-way protocol that transmits information about the capabilities and current status of a device and its interfaces. LLDP devices use the protocol to solicit information only from other LLDP devices.

LLDP supports a set of attributes that it uses to discover other devices. These attributes contain type, length, and value (TLV) descriptions. LLDP devices can use TLVs to send and receive information to other devices on the network. Details such as configuration information, device capabilities, and device identity can be advertised using this protocol.

LLDP advertises the following TLVs by default:

- DCBXP
- Management address
- Port description
- Port VLAN

- System capabilities
- System description
- System name

About DCBXP

The Data Center Bridging Exchange Protocol (DCBXP) is an extension of LLDP. It is used to announce, exchange, and negotiate node parameters between peers. DCBXP parameters are packaged as DCBXP TLVs in the LLDP packet. If CEE is used, DCBXP will use an acknowledgment mechanism over LLDP. When the port comes up, DCBX TLVs are sent and any DCBX TLVs received are processed. By default, the DCBX protocol is set to auto-detect, and the latest protocol version supported by both the peers is used.

Features that need to exchange and negotiate parameters with peer nodes using DCBXP are as follows:

- **Priority-based Flow Control (PFC)**—PFC is an enhancement to the existing Pause mechanism in Ethernet. It enables Pause based on user priorities or classes of service. A physical link that is divided into eight virtual links with PFC provides the capability to use Pause on a single virtual link without affecting traffic on the other virtual links. Enabling Pause on a per-user-priority basis allows administrators to create lossless links for traffic requiring no-drop service while retaining packet-drop congestion management for IP traffic.
- **Enhanced Transmission Selection (ETS)**—ETS enables optimal bandwidth management of virtual links. ETS is also called priority grouping. It enables differentiated treatments within the same priority classes of PFC. ETS provides prioritized processing based on bandwidth allocation, low latency, or best effort, resulting in per-group traffic class allocation. For example, an Ethernet class of traffic may have a high-priority designation and a best effort within that same class. ETS allows differentiation between traffic of the same priority class, thus creating priority groups.
- **Application Priority Configuration**—Carries information about the priorities that are assigned to specific protocols.
- **Priority to DSCP Mapping**—The mapping of the DSCP and COS values configured in the QoS policy are sent in the Application Priority TLV.



Note For information on the quality of service (QoS) features, see the *Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide*.

DCBXP is enabled by default, provided LLDP is enabled. When LLDP is enabled, DCBXP can be enabled or disabled using the **[no] lldp tlv-select dcbxp** command. DCBXP is disabled on ports where LLDP transmit or receive is disabled.

Beginning with Cisco NX-OS Release 10.2(3)F, an additional command is introduced: **[no] lldp tlv-select dcbxp egress-queuing**. While the **[no] lldp tlv-select dcbxp** command sends input queuing parameters in the ETS information that is exchanged with the peer, the **[no] lldp tlv-select dcbxp egress-queuing** command sends output queuing parameters in the ETS information. Hence, the bandwidths and priority information are extracted from the output queuing policy and exchanged with the peer.

At a time, you can configure either egress queuing or ingress queuing by running either **lldp tlv-select dcbxp egress-queuing** or **lldp tlv-select dcbxp** command as they overwrite each other.

The no forms of both the commands stop the DCBXP exchange on all interfaces.

To view which of the above two commands is enabled, run the **show lldp tlv-select** command.

When the default input queuing policy at system level is detached, the DCBXP exchange on all interfaces will stop sending ETS configuration and recommendation TLVs. However, the default output queuing policy at the system level cannot be detached.

High Availability

The LLDP feature supports stateless and stateful restarts. After a reboot or supervisor switchover, the running configuration is applied.

For more information on high availability, see the *Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*.

Virtualization Support

One instance of LLDP is supported.

Guidelines and Limitations for LLDP

LLDP has the following configuration guidelines and limitations:

- The LLDP feature must be enabled on the device before you can enable or disable the LLDP protocol on any interfaces.
- LLDP is supported only on physical interfaces.
- LLDP can discover up to one device per port.

Beginning with Release 10.1(1), multiple LLDP neighbors per physical interface are supported on the following platforms:

- N9K-C93180YC-FX3S
- N9K-C93108TC-FX3P
- N9K-C93180YC-FX3
- DCBXP is supported on the following platforms:
 - Cisco Nexus 9200, 9300-EX, 9300-FX, 9300-FX2 and 9300-FX3 Series switches
 - Cisco Nexus 9332C, 9332PQ, 9364C, 9372PX, 9372PX-E, and 9396PX switches
 - Cisco Nexus 9504 and 9508 switches with X9432PQ, X9464PX, X9536PQ, X9564PX, X9636PQ, X9732C-EX, and X9736C-FX, line cards
- The Cisco Nexus 3232C and 3264Q switches do not support DCBXP.
- DCBXP incompatibility messages might appear when you change the network QoS policy on a switch, if a physical cable loopback connection is present on the device. The incompatibility exists for only a short time and then resolves on its own.

- PFC TLVs are sent when pause is enabled for at least one COS value in a network-qos policy and the priority-flow-control mode is set to auto at the interface level.
- Beginning with Cisco NX-OS Release 10.2(3)F, the **[no] lldp tlv-select dcbxp egress-queuing** command is introduced to provide you the option to advertise egress queuing configuration in the switch. This feature is supported on Cisco Nexus 9200, 9300-EX, and 9300-FX platform switches.
- DCBX TLVs are sent when ingress queuing is applied if **lldp tlv-select dcbxp** command is used and when output queuing is applied if **lldp tlv-select dcbxp egress-queuing** is used.
- Beginning with Cisco NX-OS Release 10.2(3)F, the Correctly Advertise LLDP Chassis-ID feature introduces a new global configuration command—**lldp chassis-id switch**—to advertise the switch chassis MAC address instead of the port MAC address, that is, all the ports will publish only the MAC address of the switch chassis. This feature is supported on all Cisco Nexus 9000 series platform switches.
- Beginning with Cisco NX-OS Release 10.3(1)F, LLDP (Multi Neighbor and Port- channel) is supported on the Cisco Nexus 9808 platform switches.
- Beginning with Cisco NX-OS Release 10.3(1)F, PFC and ECN are supported on Cisco Nexus 9300-FX3 platform switches. On 9300-FX3 switches operating in FEX mode, DCBX is supported with ingress queuing.
- Beginning with Cisco NX-OS Release 10.3(2)F, new optional parameters such as 802.1 link aggregation, 802.1 vlan name, and 802.3 max frame size are enabled by default for the **lldp tlv-select** command.
The **VLAN name TLV** only has the first 32 characters of the VLAN name, per the standards. Only the names of 25 active VLANs are displayed.
- Beginning with Cisco NX-OS Release 10.4(2)F, LLDP is supported on Cisco Nexus 9232E-B1 platform switch.
- The system name in LLDP can also display the domain name if it is configured. If the management VRF has a domain-name configured, then that is given higher priority over the default VRF domain-name. If no domain-name exists for the management VRF, then we check for default VRF domain-name and use that if its configured.

Default Settings for LLDP

This table lists the LLDP default settings.

Parameters	Default
Global LLDP	Disabled
LLDP on interfaces	Enabled, after LLDP is enabled globally
LLDP hold time (before discarding)	120 seconds
LLDP reinitialization delay	2 seconds
LLDP timer (packet update frequency)	30 seconds
LLDP TLVs	Enabled
LLDP receive	Enabled, after LLDP is enabled globally
LLDP transmit	Enabled, after LLDP is enabled globally

Parameters	Default
DCBXP	Enabled, provided LLDP is enabled
DCBXP version	Auto-detect

Configuring LLDP



Note Cisco NX-OS commands for this feature may differ from Cisco IOS commands for a similar feature.

Enabling or Disabling LLDP Globally

You can enable or disable LLDP globally on a device. You must enable LLDP globally to allow a device to send and receive LLDP packets.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] feature lldp Example: switch(config)# feature lldp	Enables or disables LLDP on the device. LLDP is disabled by default. Note When this command is enabled, by default, the switch advertises the port MAC address per port.
Step 3	(Optional) [no] lldp chassis-id switch Example: switch(config)# lldp chassis-id switch	Enable this command to indicate that the switch chassis MAC address must be advertised for all the ports. Use the no form of this command to revert to advertising the port MAC address per port. Note Use the show vdc detail command to view the switch chassis MAC address.
Step 4	(Optional) show running-config lldp Example: switch(config)# show running-config lldp	Displays the global LLDP configuration. If LLDP is enabled, it shows "feature lldp." If LLDP is disabled, it shows an "Invalid command" error.

	Command or Action	Purpose
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Enabling or Disabling LLDP on an Interface

After you globally enable LLDP, it is enabled on all supported interfaces by default. However, you can enable or disable LLDP on individual interfaces or selectively configure an interface to only send or only receive LLDP packets.

Before you begin

Make sure that you have globally enabled LLDP on the device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface interface slot/port Example: <pre>switch(config)# interface ethernet 7/1 switch(config-if)#</pre>	Specifies the interface on which you are enabling LLDP and enters the interface configuration mode.
Step 3	[no] lldp transmit Example: <pre>switch(config-if)# lldp transmit</pre>	Enables or disables the transmission of LLDP packets on an interface. After you globally enable LLDP, it is enabled on all supported interfaces by default.
Step 4	[no] lldp receive Example: <pre>switch(config-if)# lldp receive</pre>	Enables or disables the reception of LLDP packets on an interface. After you globally enable LLDP, it is enabled on all supported interfaces by default.
Step 5	(Optional) show lldp interface interface slot/port Example: <pre>switch(config-if)# show lldp interface ethernet 7/1</pre>	Displays the LLDP configuration on the interface.

	Command or Action	Purpose
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring DCBXP Egress Queuing

Use the following procedure to configure DCBXP egress queuing.

Before you begin

- Make sure that you have globally enabled LLDP on the device (global configuration command **feature lldp**).



Note After you globally enable LLDP, it is enabled on all supported interfaces by default.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	Required: lldp tlv-select dcbxp Example: <pre>switch(config)# lldp tlv-select dcbxp switch(config)#</pre>	Enables DCBXP TLVs globally and starts sending input queuing parameters in the ETS information exchanged with the peer.
Step 3	(Optional) lldp tlv-select dcbxp egress-queuing Example: <pre>switch(config)# lldp tlv-select dcbxp egress-queuing switch(config)#</pre>	Enables DCBXP TLVs globally and starts sending output queuing parameters in the ETS information.

Configuring the DCBXP Protocol Version

You can specify the protocol version in which the DCBX TLVs are sent.

**Note**

- Nexus 9000 switches do not support auto-negotiating or hard coding DCBXP TLV using CIN version.
- If the peers are not running the same version, DCBX parameters may not converge for the link. Reset the link for the new protocol version to take effect.

Before you begin

Make sure that you have globally enabled LLDP on the device.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface interface slot/port Example: <pre>switch(config)# interface ethernet 1/25 switch(config-if)#</pre>	Enters interface configuration mode.
Step 3	lldp dcbx version cee/ieee/auto Example: <pre>switch(config-if)#lldp dcbx version cee</pre>	Specifies the protocol version mode sent. • The <i>cee</i> variable sets the port to only send TLVs in Converged Enhanced Ethernet (CEE) protocol version. • The <i>ieee</i> variable sets the port to only sent TLVs in IEEE 802.1Qaz protocol version. • The <i>auto</i> variable sets the port to send TLVs in the latest protocol version supported by both the peers. The default is set to <i>auto</i> . Note Devices that do not support IEEE 802.1Qaz may not properly respond to auto-negotiation attempts and may require the interface to be manually configured for lldp dcbx version cee.

Multiple LLDP Neighbors Per Physical Interface

Often times a network device sends multiple LLDP packets, out of which one is from the actual host. If a Cisco Nexus switch is communicating with the device but can only manage a single LLDP neighbor per

interface, there is a good chance that becoming a neighbor with the actual required host will fail. To minimize this, Cisco Nexus switch interfaces can support multiple LLDP neighbors creating a better opportunity of becoming an LLDP neighbor with the correct device.

Support for multiple LLDP neighbors over the same interface requires LLDP multi-neighbor support to be configured globally.



Note You must disable DCBX globally before configuring LLDP multi-neighbor support. Failure to do so invokes an error message.

Enabling or Disabling LLDP Multi-Neighbor Support

Before you begin

Consider the following before enabling LLDP multi-neighbor support on the interfaces:

- Make sure that you have globally enabled LLDP on the device (global configuration command **feature lldp**).



Note After you globally enable LLDP, it is enabled on all supported interfaces by default.

- A maximum of three (3) neighbors are supported on an interface.
- LLDP multi-neighbor is not supported on FEX interfaces.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode. Note If the output of the show lldp tlv-select command shows dcbxp, then perform step 2 and skip step 3. If the output shows dcbxp egress-queuing then skip step 2 and perform step 3. This is required to avoid invoking an error message when you configure LLDP multi-neighbor support.
Step 2	Required: no lldp tlv-select dcbxp Example: <pre>switch(config)# no lldp tlv-select dcbxp switch(config)#</pre>	Disables DCBXP TLVs globally.

	Command or Action	Purpose
Step 3	Required: no lldp tlv-select dcbxpegress-queuing Example: switch(config)# no lldp tlv-select dcbxpegress-queuing switch(config)#	Disables DCBXP TLVs globally.
Step 4	Required: [no] lldp multi-neighbor Example: switch(config)# lldp multi-neighbor switch(config)#	Enables or disables LLDP multi-neighbor support for all interfaces globally.
Step 5	interface port / slot Example: switch(config)# interface 1/1 switch(config-if)#	Specifies the interface on which you are enabling LLDP and enters the interface configuration mode.
Step 6	(Optional) [no] lldp transmit Example: switch(config-if)# lldp transmit	Disables (or enables) the transmission of LLDP packets on the interface. Note The transmission of LLDP packets on this interface was enabled using the global feature lldp command. This option is to disable the feature for this specific interface.
Step 7	(Optional) [no] lldp receive Example: switch(config-if)# lldp receive	Disables (or enables) the reception of LLDP packets on the interface. Note The reception of LLDP packets on this interface was enabled using the global feature lldp command. This option is to disable the feature for this specific interface.
Step 8	(Optional) show lldp interface port / slot Example: switch(config-if)# show lldp interface 1/1	Displays the LLDP configuration on the interface.
Step 9	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Enabling or Disabling LLDP Support on Port-Channel Interfaces

Before you begin

Consider the following before enabling LLDP support on port-channels:

- Make sure that you have globally enabled LLDP on the device (global configuration command **feature lldp**).



Note After you globally enable LLDP, it is enabled on all supported interfaces by default.

- Applying the **lldp transmit** and **lldp receive** configuration commands to a port-channel does not affect the configuration for the members of the port-channel.
- LLDP neighbors form between the port-channels only when LLDP transmit and receive is configured on both sides of the port-channel.
- The LLDP transmit and receive commands do not work on MCT, VPC, fex-fabric, FEX port-channels, and port-channel sub-interfaces.



Note If you enable the LLDP port-channel feature globally, the LLDP configuration is not applied to any of these port types. If the configuration is removed from the port-channels or the port type feature is disabled globally, you cannot use the **lldp port-channel** command to enable it on the newly supported port-channels. The command was already issued. To enable LLDP port-channel on the port-channels in question, configure **lldp transmit** and **lldp receive** for each port-channel (see steps 4, 5, and 6 in the following procedure).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode. Note If the output of the show lldp tlv-select command shows dcbxp, then perform step 2 and skip step 3. If the output shows dcbxp egress-queuing then skip step 2 and perform step 3. This is required before configuring LLDP on port-channels.
Step 2	Required: no lldp tlv-select dcbxp Example:	Disables DCBXP TLVs globally.

	Command or Action	Purpose
	<pre>switch(config)# no lldp tlv-select dcbxp switch(config)#</pre>	
Step 3	Required: no lldp tlv-select dcbxp egress-queuing Example: <pre>switch(config)# no lldp tlv-select dcbxp egress-queuing switch(config)#</pre>	Disables DCBXP TLVs globally.
Step 4	Required: [no] lldp port-channel Example: <pre>switch(config)# lldp port-channel switch(config)#</pre>	Enables or disables LLDP transmit and receive for all port channels globally.
Step 5	interface port-channel [<i>port-channel-number</i> <i>port-channel-range</i>] Example: <pre>switch(config)# interface port-channel 3 switch(config-if)#</pre> Example: Enter a range of port-channel numbers if you are configuring LLDP over more than one port-channel: <pre>switch(config)# interface port-channel 1-3 switch(config-if-range)#</pre>	Specifies the interface port-channel on which you are enabling LLDP and enters the interface configuration mode. Specifies the interface port-channel range on which you are enabling LLDP and enters the interface range configuration mode.
Step 6	(Optional) [no] lldp transmit Example: <pre>switch(config-if)# lldp transmit</pre>	Disables (or enables) the transmission of LLDP packets on the port-channel or range of port-channels. Note The transmission of LLDP packets on this port-channel was enabled using the global lldp port-channel command in step 3. This option is to disable the feature for this specific port-channel.
Step 7	(Optional) [no] lldp receive Example: <pre>switch(config-if)# lldp receive</pre>	Disables (or enables) the reception of LLDP packets on the port-channel or range of port-channels. Note The reception of LLDP packets on this port-channel was enabled using the global lldp port-channel command in step 3. This option is to disable the feature for this specific port-channel.

	Command or Action	Purpose
Step 8	(Optional) show lldp interface port-channel <i>port-channel-number</i> Example: <pre>switch(config-if)# show lldp interface port-channel 3</pre>	Displays the LLDP configuration on the port-channel.
Step 9	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring Optional LLDP Parameters

You can configure the frequency of LLDP updates, the amount of time for a receiving device to hold the information before discarding it, and the initialization delay time. You can also select the TLVs to include in LLDP packets.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	(Optional) [no] lldp holdtime <i>seconds</i> Example: <pre>switch(config)# lldp holdtime 200</pre>	Specifies the amount of time in seconds that a receiving device should hold the information that is sent by your device before discarding it. The range is 10 to 255 seconds; the default is 120 seconds.
Step 3	(Optional) [no] lldp reinit <i>seconds</i> Example: <pre>switch(config)# lldp reinit 5</pre>	Specifies the delay time in seconds for LLDP to initialize on any interface. The range is 1 to 10 seconds; the default is 2 seconds.
Step 4	(Optional) [no] lldp timer <i>seconds</i> Example: <pre>switch(config)# lldp timer 50</pre>	Specifies the transmission frequency of LLDP updates in seconds. The range is 5 to 254 seconds; the default is 30 seconds.
Step 5	(Optional) show lldp timers Example: <pre>switch(config)# show lldp timers</pre>	Displays the LLDP hold time, delay time, and update frequency configuration.

	Command or Action	Purpose
Step 6	(Optional) [no] lldp tlv-select tlv Example: <pre>switch(config)# lldp tlv-select system-name switch(config)# lldp tlv-select port link-aggregation</pre>	Specifies the TLVs to send and receive in LLDP packets. The available TLVs are management-address, port-description, port-vlan, system-capabilities, system-description, port vlan-name, port link-aggregation, port max-framesize, and system-name. All available TLVs are enabled by default. Use the no form of the command to disable the TLV.
Step 7	(Optional) show lldp tlv-select Example: <pre>switch(config)# show lldp tlv-select</pre>	Displays the LLDP TLV configuration.
Step 8	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Verifying the LLDP Configuration

To display the LLDP configuration, perform one of the following tasks:

Command	Purpose
show running-config lldp	Displays the global LLDP configuration.
show lldp interface interface slot/port	Displays the LLDP interface configuration.
show lldp timers	Displays the LLDP hold time, delay time, and update frequency configuration.
show lldp tlv-select	Displays the LLDP TLV configuration.
show lldp neighbors {detail interface interface slot/port}	Displays the LLDP neighbor device status. Note If the neighboring switch advertises switch MAC, then this show command displays the switch MAC and if it advertises the port MAC, the show command will display the port MAC.
show lldp traffic	Displays the LLDP counters, including the number of LLDP packets sent and received by the device, the number of discarded packets, and the number of unrecognized TLVs.

Command	Purpose
show lldp traffic interface <i>interface slot/port</i>	Displays the number of LLDP packets sent and received on the interface.
show qos dcbxp <i>interface slot/port</i>	Displays DCBXP information for a specific interface.

Use the **clear lldp counters** command to clear the LLDP statistics.

Configuration Example for LLDP

This example shows how to enable LLDP on a device; disable LLDP on some interfaces; configure optional parameters such as hold time, delay time, and update frequency; and disable several LLDP TLVs:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# feature lldp
switch(config)# interface ethernet 7/9
switch(config-if)# no lldp transmit
switch(config-if)# no lldp receive
switch(config-if)# exit
switch(config)# interface ethernet 7/10
switch(config-if)# no lldp transmit
switch(config-if)# no lldp receive
switch(config-if)# exit
switch(config)# lldp holdtime 200
switch(config)# lldp reinit 5
switch(config)# lldp timer 50
switch(config)# no lldp tlv-select port-vlan
switch(config)# no lldp tlv-select system-name
```




CHAPTER 25

Configuring NetFlow

This chapter describes how to configure the NetFlow feature on Cisco NX-OS devices.

This chapter contains the following sections:

- [About NetFlow, on page 425](#)
- [Prerequisites for NetFlow, on page 428](#)
- [Guidelines and Limitations for NetFlow, on page 428](#)
- [Configure Ingress NetFlow, on page 432](#)
- [Ingress NetFlow Verification Commands, on page 442](#)
- [Monitoring NetFlow, on page 444](#)
- [Display Example for NetFlow, on page 444](#)
- [Configuration Example for NetFlow, on page 444](#)

About NetFlow

NetFlow identifies packet flows for ingress IP packets and provides statistics based on these packet flows. NetFlow does not require any change to either the packets themselves or to any networking device.

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 (or VLAN) 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.

Cisco NX-OS 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 NetFlow cache.

You can export the data that NetFlow gathers for your flow by using a flow exporter and export this data to a remote NetFlow Collector, such as Cisco Stealthwatch. Cisco NX-OS exports a flow as part of a NetFlow export User Datagram Protocol (UDP) datagram under the following circumstances:

- Flows are exported periodically as per the flow timeout value, which defaults to 10 seconds if not configured.
- You have forced the flow to export.

The flow record determines the size of the data to be collected for a flow. The flow monitor combines the flow record and flow exporter with the NetFlow cache information.

Cisco NX-OS can gather NetFlow statistics and analyze all packets on the interface or subinterface.

Dual-Layer NetFlow Implementation

Unlike other Cisco Nexus platforms, Cisco Nexus 9000 Series switches separate NetFlow processing into two layers:

- The first layer supports per-packet visibility for line-rate traffic. Packets do not need to be sampled and statistically analyzed. Instead, the packets can be processed and aggregated at line rate.
- The second layer enables the gathering of flows at scale. It can maintain hundreds of thousands of flows without losing any flows and periodically exports them to an external collector.

Flow Records

A flow record defines the keys that NetFlow uses to identify packets and other fields of interest that NetFlow gathers for the flow. You can define a flow record with any combination of keys and fields of interest. Cisco NX-OS supports a rich set of keys. A flow record also defines the types of counters gathered per flow. You can configure 32- or 64-bit packet or byte counters.

The key fields are specified with the **match** keyword. The fields of interest and counters are specified under the **collect** keyword.

Cisco NX-OS enables the following match fields as the defaults when you create a flow record:

- match interface input
- match flow direction

Flow Exporters

A flow exporter contains network layer and transport layer details for the NetFlow export packet. You can configure the following information in a flow exporter:

- Export destination IP address
- Source interface
- UDP port number (where the NetFlow Collector is listening for NetFlow packets)—The default value is 9995.



Note NetFlow export packets use the IP address that is assigned to the source interface. If the source interface does not have an IP address assigned to it, the flow exporter drops flows that were meant to be exported. The Netflow Exporter source interface and destination IP must use the same VRF.

Cisco NX-OS exports data to the NetFlow Collector whenever a timeout occurs. You can configure a flush cache timeout (using the **flow timeout** command) to flush the cache and force a flow export.

Export Format

Cisco NX-OS supports the Version 9 export format. This format supports a more efficient network utilization than the older Version 5 export format and supports IPv6 and Layer 2 fields. In addition, the Version 9 export format supports the full 32-bit SNMP `ifIndex` values at the NetFlow Collector.

Layer 2 NetFlow Keys

You can define Layer 2 keys in flexible NetFlow records that you can use to capture flows in Layer 2 interfaces. The Layer 2 keys are as follows:

- Source and destination MAC addresses
- Source VLAN ID
- EtherType from the Ethernet frame

You can apply Layer 2 NetFlow to the following interfaces for the ingress direction:

- Switch ports in access mode
- Switch ports in trunk mode
- Layer 2 port channels

**Note**

You cannot apply Layer 2 NetFlow to VLANs, egress interfaces, or Layer 3 interfaces such as VLAN interfaces.

Flow Monitors

A flow monitor references the flow record and flow exporter. You apply a flow monitor to an interface.

NetFlow Output Interface

The NetFlow output interface on Cisco Nexus switches have the following features:

- NetFlow in the **show flow cache** command displays `output_if_id` and exports output interface to the collector.
- The NetFlow output interface for Cisco Nexus platform switches supports both IPv4 and IPv6 traffic flows. However, the NetFlow output interface for Cisco Nexus 9500 platform switches is supported only for IPv4 traffic flows and is not supported for IPv6 traffic flows.
- The **show flow cache** command displays `output_if_id` as 0x0. However, beginning with Cisco NX-OS Release 10.3(3)F, the **show flow cache** command displays `output_if_id` as 0x0 only if the destination IP address cannot be resolved or is not installed in the routing table or the packet received is control packet.
- NetFlow supports exporting output interface to the collector for IPv4/IPv6 incoming traffic flows, which have Next-Hop as destination interface. The NetFlow export format for `InputInt` and `OutputInt` support the full 32-bit SNMP `ifIndex` values at the NetFlow Collector.

- The NetFlow output interface is not supported for tunnel traffic flows such as MPLS, VXLAN, and GRE.
- For more information on examples for NetFlow output interface, see the [Display Example for NetFlow, on page 444](#).

High Availability

Cisco NX-OS supports stateful restarts for NetFlow. After a reboot, Cisco NX-OS applies the running configuration.

The flow cache is not preserved across restarts, and packets that come to the software during restarts cannot be processed.

Prerequisites for NetFlow

NetFlow has the following prerequisites:

- Make sure that you understand the resources required on your device because NetFlow consumes memory and CPU resources.

Guidelines and Limitations for NetFlow



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

NetFlow has the following configuration guidelines and limitations:

- For Cisco Nexus 9300-FX platform switches only, if you add a member to a port channel that is already configured for Layer 2 NetFlow, its NetFlow configuration is removed and the Layer 2 configuration of the port channel is added to it.
- NetFlow is not supported on tunnel interfaces.
- NetFlow is not supported for CPU-transmitted packets.
- Only ingress NetFlow is supported. Egress NetFlow is not supported.
- Flow cache can be cleared per flow type, such as Layer 2, IPv4, and IPv6. It cannot be cleared per flow monitor.
- On Nexus 9000 Switches, NetFlow collects ICMP flow information and sends it to the collector and the ICMP type and code are inherently included in the packets. ICMP packets in NetFlow or flow export records do not use traditional source and destination ports like TCP or UDP. Instead, the exporter often encodes the ICMP type and code into fields like SPORT (source port) and DPORT (destination port). For example:
 - ICMP Echo Request: SPORT=2048, DPORT=0
 - ICMP Echo Reply: SPORT=0, DPORT=0

- ICMP Time Exceeded: SPORT=2816, DPORT=0
- Flow collection is not performed for ARP traffic.
- You must configure a source interface for the NetFlow Data Export (NDE). If you do not configure a source interface, the flow exporter drops flows that were meant to be exported.
- Layer 2 switched flow monitors are applied only to Layer 2 interfaces. IP and IPv6 flow monitors can be applied to VLANs, SVIs, Layer 3 routed interfaces, or subinterfaces.
- If you change a Layer 2 interface to a Layer 3 interface, or a Layer 3 interface to a Layer 2 interface, the software removes the Layer 2 NetFlow configuration from the interface.
- The same flow monitor cannot be shared with a VLAN and Layer 3 interfaces (for example, physical Layer 3 interface, SVI interface, or Layer 3 subinterface). You must distinguish a VLAN and Layer 3 interface since the ACL is different and cannot be shared. They must be treated as two different profiles.
- A rollback fails if you try to modify a record that is programmed in the hardware during a rollback.
- The limitations of the NetFlow feature are as follows:
 - NetFlow for MPLS/VXLAN datapath is not supported
 - NetFlow is not supported on loopback and switch management interfaces.
- The following guidelines and limitations are applicable to Netflow in a VXLAN environment:
 - NetFlow is supported on SVI and non-uplink L3 Interfaces of a VXLAN VTEP. This does not include the L3VNI SVI.
 - NetFlow is not supported on uplink interfaces on a VXLAN VTEP.
 - NetFlow on Multisite Border Gateways is not supported.
 - A NetFlow Collector that is reachable over the VXLAN fabric is supported.
- Beginning with Cisco NX-OS Release 9.2(1):
 - NetFlow for FEX Layer 3 ports is supported on Cisco Nexus 9300-EX and 9300-FX platform switches.
 - NetFlow CE is supported on the Cisco Nexus 9300-EX platform switches.

**Note**

All EX type platform switches, including the Cisco Nexus 9700-EX line cards, CE NetFlow only captures CE flow records for non-IPv4 and IPv6 traffic flows. Whereas for FX and FX2 type platform switches and line cards, we can capture CE flow data for IP flows as long as **mac packet-classify** is applied on the interface.

- Beginning with Cisco NX-OS Release 9.2(2), the Cisco Nexus 9300-FX switch supports collecting the OUTPUT_SNMP field for NetFlow Data Export (NDE). No other Cisco Nexus 9000 platform switch or Cisco Nexus line card supports collecting the OUTPUT_SNMP field.

- Beginning with Cisco NX-OS Release 9.2(2), NetFlow is supported on Cisco Nexus 9500 platform switches with Cisco Nexus 9700-EX line cards and FM-E modules.
- NetFlow is not supported on Cisco Nexus 92348GC-X platform switch.
- For Cisco Nexus 9300-EX platform switches, a flow monitor applied on a VLAN or SVI can collect flows for both switched and routed traffic. For Cisco Nexus 9300-FX platform switches, NetFlow VLANs are supported for switched traffic only, and NetFlow SVIs are supported for routed traffic only.
- The Cisco Nexus 9300-EX platform switch supports NetFlow and SPAN on the same interface at the same time. This functionality is a viable alternative to using SPAN and sFlow.
- On Cisco Nexus 9300-EX/FX platform switches, and Cisco Nexus 9500 platform switches with EX/FX modules, SPAN, and sFlow cannot both be enabled simultaneously. If one is active, the other cannot be enabled. However, on the Cisco Nexus 9300-EX/FX/FX2 and the Cisco Nexus 9500 platform switches with EX modules, both NetFlow and SPAN can be enabled simultaneously, providing a viable alternative to using sFlow and SPAN.



Note Cisco Nexus 9300-FX2 platform switches support sFlow and SPAN coexistence.

- For Cisco Nexus 9300-EX platform switches, the same flow monitor cannot be attached to a VLAN and an SVI at the same time.
- The Cisco Nexus 9300-EX platform switches have dedicated TCAM and do not require carving.
- TCAM carving configuration of the ing-netflow region can be performed on FX line cards. EX line cards have a default ing-netflow region TCAM carving of 1024 and cannot be configured otherwise. For ports on the EX and FX line cards, the suggested maximum for the ing-netflow region is 1024.
- The ToS field is not exported for Cisco Nexus 9300-EX platform switches.
- Record match that is based on IP ToS, is not supported for IPv6 flow monitors. The ToS value is collected on the collector as 0x0 irrespective of the value the traffic holds.

This limitation is applicable for the following platform switch families:

- Cisco Nexus 9300-EX
 - Cisco Nexus 9300-FX
 - Cisco Nexus 9300-FX2
 - Cisco Nexus 9300-FX3
 - Cisco Nexus 9300-GX
 - Cisco Nexus 9500 with EX and FX line cards
- The following guideline applies to all Cisco Nexus 9500 platform switches with EX and FX line cards: Configuring an EX port as a trunk when FX ports are trunks with NetFlow configurations already applied, does not remove the unsupported EX NetFlow configuration from the FX port trunks. For example, if you apply more than two different IPv4 flow monitors to FX port trunks and if EX ports are added to the same trunks, the configuration on the trunks beyond the two monitors is not automatically removed, since it's only an EX port limitation. Since this configuration will not report flows beyond two monitors

for EX trunk ports, we recommend that you use only two monitors per protocol (v4/v6/CE) on modular switches that could potentially have both EX and FX ports in the same trunk.

-
- Commands **record netflow ipv4 original-input**, **record netflow ipv4 original-output**, and **record netflow layer2-switched input** are not supported in Cisco NX-OS Release 9.3(1).
- Beginning with Cisco NX-OS Release 9.3(3), the following Non-Disruptive In-Service Software Upgrade (ND ISSU) limitations about NetFlow apply for all Cisco Nexus 9000 Series switches:
 - While performing an ND ISSU, a two-minute export loss is expected.
 - During an ND ISSU, an exporter with a management interface source port is not supported. Export loss is expected until the management interface comes up.
- Beginning with Cisco NX-OS Release 9.3(3), ingress NetFlow is supported on Cisco Nexus 9300-GX platform switch.
- Beginning with Cisco NX-OS Release 9.3(4), the following RTP/NetFlow monitoring limitation exists:

The RTP monitoring feature enables a monitor of RTP flows on all interfaces of a switch and reports them in the **show flow rtp detail** command output. An RTP flow is any UDP flow with a source port within the range of 16384-32767. If a NetFlow monitor is attached to a switch interface with RTP monitoring enabled, then all the traffic/flows (including the RTP flows) on that interface are reported in the output of the **show flow cache** command. The RTP flows will no longer be shown in the output of the **show flow rtp detail** command. When the attached monitor is removed, the RTP flows are reported again in the **show flow rtp detail** command output.

This limitation impacts the following switches:

 - Cisco Nexus 9336C-FX2
 - Cisco Nexus 93240YC-FX2
 - Cisco Nexus 9348GC-FXP
 - Cisco Nexus 93180YC-FX
 - Cisco Nexus 93108TC-FX
 - Cisco Nexus 9316D-GX
 - Cisco Nexus 93600CD-GX
 - Cisco Nexus 9364C-GX
 - Cisco Nexus 9504, 9508, and 9516 with the 9736C-FX line card
- Cisco Nexus 9500 platform switches with FM-E, FM-E2, and FM-E3 modules and Cisco Nexus 9300-FX/FX3 switches support the NetFlow output interface feature. However, output interface is not supported on 9300-EX and 9500-EX platform switches.
- NetFlow is supported on Cisco Nexus 9500 platform switches with EX, FX, and GX mixed chassis. You can use SPAN simultaneously with NetFlow on the Cisco Nexus 9500 platform switches with EX, FX, and GX mixed chassis. Cisco Nexus 9500-GX platform switches does not support SPAN with sFlow feature mix.
- The Cisco Nexus 3232C and 3264Q switches do not support NetFlow.

- Beginning with Cisco NX-OS Release 10.1(2), Netflow is supported on N9K-X9716D-GX line card.
- Enable NetFlow only on platforms that support this feature.
- The **match ip tos** command is present in flow record configuration options, but the functionality is not supported.
- Beginning with Cisco NX-OS Release 10.2(1)F, Layer 3 NetFlow on Layer 2 interfaces is supported on Cisco Nexus 9300-EX, 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, and 9300-GX2 platform switches, and on 9500-EX LC and 9500-FX LC. Few guidelines and limitations are as follows:
 - You can attach either Layer 3 flow monitor or Layer 2 flow monitor to Layer 2 interface, not both.
 - If a flow monitor is already attached to Layer 3 interface, then the same flow monitor cannot be attached to Layer 2 interface.
 - The **mac-packet-classify** command is not supported, when Layer 3 flow monitor is applied on Layer 2 interface.
- Beginning with Cisco NX-OS Release 10.3(3)F, Netflow Output_if_id is supported on Cisco Nexus 9300-FX2, 9300-GX, and 9300-GX2 switches and 9500-GX Line Cards.
 - For unicast flows, if the output_if_id is shown as 0x0, it means that the destination IP is not resolved or not available in the routing table or the ARP table.
 - The output_if_id is shown as 0x0 for received packets destined to the control-plane. Examples of such traffic are routing protocols or ICMP packets.
 - For IPv6 flows on Cisco Nexus 9500 Line Cards, output_if_id is shown as Unsupported due to hardware limitations.
 - The output_if_id for data plane multicast traffic will be shown as 0x0.

**Note**

For verified NetFlow scalability numbers, see the [Cisco Nexus 9000 Series NX-OS Verified Scalability Guide](#).

Configure Ingress NetFlow

Procedure

-
Step 1	Enable the NetFlow feature.
Step 2	Define a flow record by specifying keys and fields to the flow.
Step 3	Define an optional flow exporter by specifying the export format, protocol, destination, and other parameters.
Step 4	Define a flow monitor based on the flow record and flow exporter.
Step 5	Apply the flow monitor to a source interface , subinterface, or VLAN interface .
-

Enable NetFlow feature

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] feature netflow Example: <pre>switch(config)# feature netflow</pre>	Enables or disables the NetFlow feature. The default is disabled. Note The Cisco Nexus 9500 platform switches with N9K-T2 EoR do not support NetFlow.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Create a flow record

You can create a flow record and add keys to match on and nonkey fields to collect in the flow.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	flow record <i>name</i> Example: <pre>switch(config)# flow record Test switch(config-flow-record)#</pre>	Creates a flow record and enters flow record configuration mode. You can enter up to 63 alphanumeric characters for the flow record name.
Step 3	(Optional) description <i>string</i> Example: <pre>switch(config-flow-record)# description IPv4Flow</pre>	Describes this flow record as a maximum 63-character string.
Step 4	(Optional) match <i>type</i> Example:	Specifies a match key. For more information, see Specify match parameters, on page 434 .

Specify match parameters

	Command or Action	Purpose
	<code>switch(config-flow-record) # match transport destination-port</code>	Note The match transport destination-port and match ip protocol commands are required to export Layer 4 port data.
Step 5	(Optional) collect type Example: <code>switch(config-flow-record) # collect counter packets</code>	Specifies the collection field. For more information, see Specify collect parameters, on page 435 .
Step 6	(Optional) show flow record <i>[name]</i> <i>[record-name]</i> { netflow-original netflow protocol-port netflow { ipv4 ipv6 } { original-input original-output }} Example: <code>switch(config-flow-record) # show flow record netflow protocol-port</code>	Displays information about NetFlow flow records. You can enter up to 63 alphanumeric characters for the flow record name.
Step 7	(Optional) copy running-config startup-config Example: <code>switch(config-flow-record) # copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Specify match parameters

You must configure at least one of the following match parameters for flow records:

Command	Purpose
match datalink { mac source-address mac destination-address ethertype vlan } Example: <code>switch(config-flow-record) # match datalink ethertype</code>	Specifies the Layer 2 attribute as a key.
match ip { protocol tos } Example: <code>switch(config-flow-record) # match ip protocol</code>	Specifies the IP protocol or ToS fields as keys. Note The match transport destination-port and match ip protocol commands are required to export Layer 4 port data. The data is collected and displayed in the output of the show hardware flow ip command but is not collected and exported until you configure both commands.

Command	Purpose
match ipv4 {destination address source address} Example: <pre>switch(config-flow-record)# match ipv4 destination address</pre>	Specifies the IPv4 source or destination address as a key.
match ipv6 {destination address source address flow-label options} Example: <pre>switch(config-flow-record)# match ipv6 flow-label</pre>	Specifies the IPv6 key.
match transport {destination-port source-port} Example: <pre>switch(config-flow-record)# match transport destination-port</pre>	Specifies the transport source or destination port as a key. Note The match transport destination-port and match ip protocol commands are required to export Layer 4 port data. The data is collected and displayed in the output of the show hardware flow ip command but is not collected and exported until you configure both commands.

Specify collect parameters

In Cisco/NX-OS NetFlow, “collect parameters” are the configuration options you specify within a flow record that indicate which fields or metrics are captured from a network flow for export and analysis.

You must configure at least one of the following collect parameters for the flow records:

Command	Purpose
collect counter {bytes packets} [long] Example: <pre>switch(config-flow-record)# collect counter packets</pre>	Collects either packet-based or byte counters from the flow. You can optionally specify that 64-bit counters are used.
collect ip version Example: <pre>switch(config-flow-record)# collect ip version</pre>	Collects the IP version for the flow.
collect timestamp sys-uptime {first last} Example: <pre>switch(config-flow-record)# collect timestamp sys-uptime last</pre>	Collects the system up time for the first or last packet in the flow.

Command	Purpose
collect transport tcp flags Example: <pre>switch(config-flow-record)# collect transport tcp flags</pre>	Collects the TCP transport layer flags for the packets in the flow.

Create a flow exporter

The flow exporter configuration defines the export parameters for a flow and specifies reachability information for the remote NetFlow Collector.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	flow exporter name Example: <pre>switch(config)# flow exporter flow-exporter-one switch(config-flow-exporter)#</pre>	Creates a flow exporter and enters flow exporter configuration mode. You can enter up to 63 alphanumeric characters for the flow exporter name.
Step 3	destination {ipv4-address ipv6-address} [use-vrf name] Example: <pre>switch(config-flow-exporter)# destination 192.0.2.1</pre>	Sets the destination IPv4 or IPv6 address for this flow exporter. You can optionally configure the VRF to use to reach the NetFlow Collector. You can enter up to 32 alphanumeric characters for the VRF name.
Step 4	source interface-type name/port Example: <pre>switch(config-flow-exporter)# source ethernet 2/1</pre>	Specifies the interface to use to reach the NetFlow Collector at the configured destination.
Step 5	(Optional) description string Example: <pre>switch(config-flow-exporter)# description exportversion9</pre>	Describes this flow exporter. You can enter up to 63 alphanumeric characters for the description.
Step 6	(Optional) dscp value Example: <pre>switch(config-flow-exporter)# dscp 0</pre>	Specifies the differentiated services codepoint value. The range is from 0 to 63.

	Command or Action	Purpose
Step 7	(Optional) transport udp port Example: <pre>switch(config-flow-exporter)# transport udp 200</pre>	Specifies the UDP port to use to reach the NetFlow Collector. The range is from 0 to 65535. Note If you do not specify the UDP port, 9995 is selected as the default.
Step 8	version 9 Example: <pre>switch(config-flow-exporter)# version 9 switch(config-flow-exporter-version-9)#</pre>	Specifies the NetFlow export version. Choose version 9 to enter the flow exporter version 9 configuration submode.
Step 9	(Optional) option {exporter-stats interface-table} timeout seconds Example: <pre>switch(config-flow-exporter-version-9)# option exporter-stats timeout 1200</pre>	Sets the flow exporter statistics resend timer. The range is from 1 to 86400 seconds.
Step 10	(Optional) template data timeout seconds Example: <pre>switch(config-flow-exporter-version-9)# template data timeout 1200</pre>	Sets the template data resend timer. The range is from 1 to 86400 seconds.
Step 11	(Optional) copy running-config startup-config Example: <pre>switch(config-flow-exporter-version-9)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Create Flow Monitor

You can create a flow monitor and associate it with a flow record and a flow exporter. All of the flows that belong to a monitor use the associated flow record to match on the different fields, and the data is exported to the specified flow exporter.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	flow monitor <i>name</i> Example: <pre>switch(config)# flow monitor flow-monitor-one switch(config-flow-monitor)#</pre>	Creates a flow monitor and enters flow monitor configuration mode. You can enter up to 63 alphanumeric characters for the flow monitor name.
Step 3	(Optional) description <i>string</i> Example: <pre>switch(config-flow-monitor)# description IPv4Monitor</pre>	Describes this flow monitor. You can enter up to 63 alphanumeric characters for the description.
Step 4	(Optional) exporter <i>name</i> Example: <pre>switch(config-flow-monitor)# export v9</pre>	Associates a flow exporter with this flow monitor. You can enter up to 63 alphanumeric characters for the exporter name.
Step 5	record <i>name</i> [netflow-original netflow protocol-port netflow {ipv4 ipv6} {original-input original-output}] Example: <pre>switch(config-flow-monitor)# record IPv4Flow</pre>	Associates a flow record with the specified flow monitor. You can enter up to 63 alphanumeric characters for the record name. Note record netflow ipv4 original-input , record netflow ipv4 original-output , and record netflow layer2-switched input are not supported in Cisco NX-OS Release 9.3(1).
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config-flow-monitor)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Apply Flow Monitor to an Interface

You can apply a flow monitor to an interface. You can apply a flow monitor in egress direction using the **output** keyword instead of **input**.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	interface <i>vlan</i> <i>vlan-id</i> Example:	Configures a VLAN interface and enters interface configuration mode.

	Command or Action	Purpose
	<pre>switch(config)# interface vlan 10 switch(config-if)#</pre>	
Step 3	ip flow monitor {ipv4 ipv6 layer-2-switched} input Example: <pre>switch(config-if)# ip flow monitor ipv4 input</pre>	Associates an IPv4, IPv6, or Layer 2-switched flow monitor to the interface for input packets.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure Bridged NetFlow on VLAN

You can apply a flow monitor to a VLAN in order to gather Layer 3 data over Layer 2 switched packets in a VLAN.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	vlan configuration <i>vlan-id</i> Example: <pre>switch(config)# vlan configuration 30 switch(config-vlan-config)#</pre>	Enters VLAN configuration mode. The VLAN ID range is from 1 to 3967 or from 4048 to 4093. Note VLAN configuration mode enables you to configure VLANs independently of their creation, which is required for VTP client support.
Step 3	{ip ipv6} flow monitor <i>name</i> Example: <pre>switch(config-vlan-config)# ip flow monitor testmonitor</pre>	Associates a flow monitor to the VLAN for input packets. You can enter up to 63 alphanumeric characters for the flow monitor name.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config-vlan-config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure Layer 2 NetFlow Keys

You can define Layer 2 keys in flexible NetFlow records that you can use to capture flows in Layer 2 interfaces.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	flow record name Example: <pre>switch(config)# flow record L2_record switch(config-flow-record)#</pre>	Enters flow record configuration mode. For more information about configuring flow records, see Create a flow record, on page 433 .
Step 3	match datalink {mac source-address mac destination-address ethertype vlan} Example: <pre>switch(config-flow-record)# match datalink ethertype</pre>	Specifies the Layer 2 attribute as a key.
Step 4	exit Example: <pre>switch(config-flow-record)# exit switch(config)#</pre>	Exits flow record configuration mode.
Step 5	interface {ethernet slot/port port-channel number} Example: <pre>switch(config)# interface Ethernet 6/3 switch(config-if#)</pre>	Enters interface configuration mode. The interface type can be a physical Ethernet port or a port channel.
Step 6	switchport Example: <pre>switch(config-if)# switchport</pre>	Changes the interface to a Layer 2 physical interface. For information on configuring switch ports, see the Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide .
Step 7	mac packet-classify Example: <pre>switch(config-if)# mac packet-classify</pre>	Forces MAC classification of packets. For more information on using this command, see the Cisco Nexus 9000 Series NX-OS Security Configuration Guide. Note You must use this command to capture flows.

	Command or Action	Purpose
Step 8	layer2-switched flow monitor <i>flow-name</i> input Example: <pre>switch(config-if)# layer2-switched flow monitor L2_monitor input</pre>	Associates a flow monitor to the switch port input packets. You can enter up to 63 alphanumeric characters for the flow monitor name.
Step 9	(Optional) show flow record netflow layer2-switched input Example: <pre>switch(config-if)# show flow record netflow layer2-switched input</pre>	Displays information about the Layer 2 NetFlow default record.
Step 10	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure Layer 3 NetFlow on Layer 2 Interfaces

You can define Layer 3 flow monitors on Layer 2 interfaces to capture Layer 3 flow information on Layer 2 interfaces.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	flow record <i>name</i> Example: <pre>switch(config)# flow record L3_record switch(config-flow-record)#</pre>	Enters flow record configuration mode. For more information about configuring flow records, see Create a flow record, on page 433 .
Step 3	interface {ethernet <i>slot/port</i> port-channel <i>number</i>} Example: <pre>switch(config)# interface Ethernet 6/3 switch(config-if#)</pre>	Enters interface configuration mode. The interface type can be a physical Ethernet port or a port channel.
Step 4	switchport Example: <pre>switch(config-if)# switchport</pre>	Changes the interface to a Layer 2 mode. For information on configuring switch ports, see the Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide .

	Command or Action	Purpose
Step 5	ip flow monitor <i>flow-name</i> input Example: <pre>switch(config-if)# ip flow monitor v41 input</pre>	Associates an IPv4 flow monitor to the switch port input packets. You can enter up to 63 alphanumeric characters for the flow monitor name.
Step 6	ipv6 flow monitor <i>flow-name</i> input Example: <pre>switch(config-if)# ipv6 flow monitor v61 input</pre>	Associates an IPv6 flow monitor to the switch port input packets. You can enter up to 63 alphanumeric characters for the flow monitor name.
Step 7	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configure NetFlow Timeouts

You can optionally configure global NetFlow timeouts that apply to all flows in the system.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	flow timeout <i>seconds</i> Example: <pre>switch(config)# flow timeout 30</pre>	Sets the flush timeout value in seconds. The range is from 5 to 60 seconds. The default value is 10 seconds.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Ingress NetFlow Verification Commands

An Ingress NetFlow verification command is a diagnostic command that

- displays the status and configuration of Ingress NetFlow features on a device,
- provides detailed information on flows, interfaces, exporters, and records, and

- helps administrators troubleshoot and validate NetFlow operations.

To display the Ingress NetFlow configuration, perform one of the following tasks:

Command	Purpose
show flow cache [ipv4 ipv6 ce]	Displays information about NetFlow IP flows. Note This command can appear to be not valid on the EOR switches and no flows can be seen. To view this command on the EOR switches, attach to the module using the attach mod x command or check this command using the slot x quoted “ show flow cache ” command where x is the module number of the ingress NetFlow.
show flow exporter [name]	Displays information about NetFlow flow exporters and statistics. You can enter up to 63 alphanumeric characters for the flow exporter name.
show flow interface [interface-type slot/port]	Displays information about NetFlow interfaces.
show flow record [name]	Displays information about NetFlow flow records. You can enter up to 63 alphanumeric characters for the flow record name.
show flow record netflow layer2-switched input	Displays information about the Layer 2 NetFlow configuration.
show running-config netflow	Displays the NetFlow configuration that is currently on your device.

Use the **show flow exporter** command to display NetFlow statistics. Use the **clear flow exporter** command to clear NetFlow flow exporter statistics.

Display Example for Ingress NetFlow

The output of the **show flow cache** command for IPv4 displays:

```
show flow cache
IPv4 Entries
SIP      DIP      BD ID   S-Port  D-Port  Protocol  Byte Count  Packet Count  TCP FLAGS
TOS  if_id  output_if_id  flowStart flowEnd
10.10.30.4  30.33.1.2  1480    30000   17998   17         683751850    471553        0x0
0x0  0x90105c8  0x1a005000    14096494  14153835
30.33.1.2  10.10.39.4  4145    30000   18998   17         43858456    30164         0x0
0x0  0x1a005000  0x1a006600    14096477  14099491
10.10.29.4  30.33.1.2  1479    30000   17998   17         683751850    471553        0x0
0x0  0x90105c7  0x1a005000    14096476  14153817
10.10.7.4   30.33.1.2  1457    30000   17998   17         683753300    471554        0x0
0x0  0x90105b1  0x1a005000    14096481  14153822
30.33.1.2  10.10.42.4  4145    30000   18998   17         95289344    65536         0x0
0x0  0x1a005000  0x1a006600    14112551  14119151
10.10.49.4  30.33.1.2  1499    30000   17998   17         683753300    471554        0x0
0x0  0x90105db  0x1a005000    14096486  14153827
```

Monitoring NetFlow

Use the **show flow exporter** command to display NetFlow statistics. Use the **clear flow exporter** command to clear NetFlow flow exporter statistics.

Display Example for NetFlow

The output of the **show flow cache** command for IPv4 displays:

```
show flow cache
IPV4 Entries
SIP      DIP      BD ID  S-Port  D-Port Protocol Byte Count  Packet Count  TCP FLAGS
TOS  if_id  output_if_id  flowStart flowEnd
10.10.30.4  30.33.1.2  1480  30000   17998  17      683751850    471553        0x0
0x0  0x90105c8  0x1a005000    14096494  14153835
30.33.1.2  10.10.39.4  4145  30000   18998  17      43858456    30164         0x0
0x0  0x1a005000  0x1a006600    14096477  14099491
10.10.29.4  30.33.1.2  1479  30000   17998  17      683751850    471553        0x0
0x0  0x90105c7  0x1a005000    14096476  14153817
10.10.7.4   30.33.1.2  1457  30000   17998  17      683753300    471554        0x0
0x0  0x90105b1  0x1a005000    14096481  14153822
30.33.1.2  10.10.42.4  4145  30000   18998  17      95289344    65536         0x0
0x0  0x1a005000  0x1a006600    14112551  14119151
10.10.49.4  30.33.1.2  1499  30000   17998  17      683753300    471554        0x0
0x0  0x90105db  0x1a005000    14096486  14153827
```

Configuration Example for NetFlow

This example shows how to configure a NetFlow exporter configuration for IPv4:

```
feature netflow
flow exporter ee
 destination 171.70.242.48 use-vrf management
 source mgmt0
 version 9
 template data timeout 20
flow record rr
 match ipv4 source address
 match ipv4 destination address
 collect counter bytes
 collect counter packets
flow monitor foo
 record rr
 exporter ee
interface Ethernet2/45
 ip flow monitor foo input
 ip address 10.20.1.1/24
 no shutdown
```



CHAPTER 26

Configuring Mixed Mode

This chapter describes how to configure the Mixed Mode (Analytics and NetFlow) feature on Cisco NX-OS devices.

This chapter contains the following sections:

- [About Mixed Mode, on page 445](#)
- [Guidelines and Limitations for Mixed Mode, on page 445](#)
- [Mixed Mode: Use Cases, on page 446](#)
- [Verifying the Mixed Mode Configuration, on page 449](#)
- [Display Example for Mixed Mode, on page 450](#)

About Mixed Mode

You can configure NetFlow and Analytics features on a switch, so both the features coexist and utilize the standard V9 export from CPU. This mode, in which both the features coexist, is called the mixed mode.



Note Until Cisco NX-OS Release 10.2(3)F, standard V9 export was supported only for NetFlow flow records. Beginning from Cisco NX-OS Release 10.2(3)F, standard V9 export was supported for Analytics also. However, NetFlow and Analytics features were mutually exclusive.

Guidelines and Limitations for Mixed Mode

The following guidelines and limitations are applicable to Mixed mode:

- Beginning with Cisco NX-OS Release 10.3(1)F, both NetFlow and Analytics can co-exist and use the standard V9 export from CPU resulting in decreased processing load on the collectors. However, this mixed mode is not supported on 9300-EX modules. Furthermore, transition to mixed mode is not possible to or from analytics mode. The applicable guidelines and limitations are as follows:
 - L2 flow monitor is not supported.
 - VRF filter is not supported.
 - ND ISSU is not supported.

- The IPv4 and IPv6 profiles are as follows:
 - IP flow monitor: 28
 - IPv6 flow monitor: 26
- Analytics record config must be a superset of all the record parameters.
- Configure system monitor before configuring any system filter/interface filter configs.
- Unconfigure system filter/interface filter configs, before unconfiguring system monitor.
- In mixed mode, two NetFlow records are exported for AN flow on EOR.
- Interface based FT is not supported for tunnel traffic flows such as MPLS, VXLAN, and GRE.
- Beginning with Cisco NX-OS Release 10.3(3)F, Ingress_VRF_ID is supported for the NetFlow and Analytics features on all Cisco Nexus 9000 switches.
The ingress vrf-id is captured, shown in **show flow cache** and sent to NetFlow collector.
When Layer 3 NetFlow is configured on a Layer 2 interface and the traffic is sent, and then the **show flow cache** command output displays the value of Ingress_VRF_ID as zero.
- Beginning with Cisco NX-OS Release 10.3(3)F, the NetFlow mixed mode is enabled by default. This reduces the TCAM space assigned to the analytics feature from a maximum of 512 entries to a maximum of 256 entries.
- Beginning with Cisco NX-OS Release 10.3(3)F, flow record is seen when it is defined in system filter, but not defined in interface filter unlike in earlier releases. In the earlier releases, if the interface filter is configured, the flow record was seen only if it was defined in the interface filter.

Mixed Mode: Use Cases

Mixed mode can be configured only from NetFlow mode. In a scenario where the switches already have feature Analytics enabled, unconfigure analytics first, configure NetFlow feature, and then transition to mixed mode.

The following are the possible use cases for mixed mode:

- Switches already deployed with feature Analytics
- Switches already deployed with feature NetFlow
- Switches that have neither feature configured

After configuring the mixed mode, use the standard V9 format to export both NetFlow and Analytics flow records from the CPU to the respective collectors.



Note Analytics data is a superset of NetFlow data. The additional analytics flow data such as flow latency, traffic burst data, payload length, TCP flags, IP flags, and packet disposition flags is communicated through Vendor Specific Fields (VSF).

Use Case: Switches Already Deployed with Feature Analytics

Unconfigure or Save feature Analytics configuration and perform the steps indicated in Use Case: Switches that have Neither Features Configured. Note that transition to mixed mode is not possible to or from Analytics mode.

Use Case: Switches Already Deployed with Feature NetFlow

Perform the following procedure for switches that already have feature netflow deployed on them:

1. Use the following command to perform team carving for mixed mode:

hardware flow-table analytics-netflow



Note This command disrupts the flow monitoring and record exports for a brief period.

2. Configure feature analytics as follows:

```
feature analytics
analytics
  flow filter telemetryFP
    ipv4 telemetryIpv4Acl
    ipv6 telemetryIpv6Acl
  flow exporter e11
    destination 10.10.20.21 v9
    transport udp 1100
    events transport udp 55
    source Ethernet1/42
  flow exporter e12
    destination 10.10.20.21 v9
    transport udp 9200
    events transport udp 555
    source Ethernet1/42
  flow record fte-record
    match ip source address
    match ip destination address
    match ip protocol
    match transport source-port
    match transport destination-port
    collect counter packets
    collect timestamp sys-uptime first
    collect timestamp sys-uptime last
  flow monitor m1
    record fte-record
    exporter-bucket-id 1 0 4095
    exporter e11
  flow monitor m2
    record fte-record
    exporter-bucket-id 1 0 2000
    exporter e11
    exporter-bucket-id 2 2001 4095
    exporter e12
  flow profile telemetryProf
    collect interval 1000
    source port 1001
  flow event fte-event1
    group drop-events
    capture buffer-drops
```

```

        capture acl-drops
        capture fwd-drops
    group packet-events
        capture tos 50
        capture ttl 50
    flow system config
        exporter-id 4
        monitor m1 input
        profile telemetryProf
        event fte-event1
        filter telemetryFP

```

Use Case: Switches that have Neither Features Configured

Configure feature netflow and then perform either the steps mentioned in Use Case: Switches Already Deployed with Feature NetFlow or the following steps:

```

feature netflow
hardware flow-table analytics-netflow
feature analytics
flow exporter e1
    destination 10.10.20.21
    transport udp 100
    source Ethernet1/42
    version 9
flow record r4
    match ipv4 source address
    match ipv4 destination address
    match ip protocol
    match transport source-port
    match transport destination-port
    collect counter bytes
    collect counter packets
    collect timestamp sys-uptime first
    collect timestamp sys-uptime last
flow record r6
    match ip protocol
    match transport source-port
    match transport destination-port
    match ipv6 source address
    match ipv6 destination address
    collect counter bytes
    collect counter packets
    collect timestamp sys-uptime first
    collect timestamp sys-uptime last
flow monitor m41
    record r4
    exporter e1
flow monitor m6
    record r6
    exporter e1
analytics
    flow filter telemetryFP
        ipv4 telemetryIpv4Acl
        ipv6 telemetryIpv6Acl
    flow exporter e11
        destination 10.10.20.21 v9
        transport udp 1100
        events transport udp 55
        source Ethernet1/42
    flow exporter e12
        destination 10.10.20.21 v9

```

```
transport udp 9200
events transport udp 555
source Ethernet1/42
flow record fte-record
match ip source address
match ip destination address
match ip protocol
match transport source-port
match transport destination-port
collect counter packets
collect timestamp sys-uptime first
collect timestamp sys-uptime last
flow monitor m1
record fte-record
exporter-bucket-id 1 0 4095
exporter e11
flow monitor m2
record fte-record
exporter-bucket-id 1 0 2000
exporter e11
exporter-bucket-id 2 2001 4095
exporter e12
flow profile telemetryProf
collect interval 1000
source port 1001
flow event fte-event1
group drop-events
capture buffer-drops
capture acl-drops
capture fwd-drops
group packet-events
capture tos 50
capture ttl 50
flow system config
exporter-id 4
monitor m1 input
profile telemetryProf
event fte-event1
filter telemetryFP

interface Ethernet1/42
ip flow monitor m41 input
ipv6 flow monitor m6 input
```

Verifying the Mixed Mode Configuration

To display the mixed mode configuration, perform one of the following tasks:

Command	Purpose
show flow cache [ipv4 ipv6]	Displays information about NetFlow IP flows. Note Beginning with Cisco NX-OS Release 10.3(3)F, this command output also displays Ingress VRF ID. The ingress vrf-id is captured, shown in show flow cache and sent to NetFlow collector.

Command	Purpose
show flow exporter <i>[name]</i>	Displays information about NetFlow/Analytics flow exporters and statistics. You can enter up to 63 alphanumeric characters for the flow exporter name.
show flow interface <i>[interface-type slot/port]</i>	Displays information about NetFlow/Analytics interfaces.
show flow record <i>[name]</i>	Displays information about NetFlow/Analytics flow records. You can enter up to 63 alphanumeric characters for the flow record name.
show running-config <i>[netflow analytics]</i>	Displays the coexisting NetFlow and Analytics configuration that is currently on your device.
show flow monitor	Displays the NetFlow/Analytics monitor configuration.
show flow system	Displays information about the Analytics system configuration.
show flow filter	Displays information about Analytics filters.
show flow profile	Displays information about the Analytics profile.
show flow event	Displays information about the Analytics events.

Display Example for Mixed Mode

The output of the **show flow cache** command displays:



Note Only 10k flows are displayed in XML output.



Note When Layer 3 NetFlow is configured on a Layer 2 interface and the traffic is sent, and then the **show flow cache** command is run, the output displays the value of Ing-VRF as zero.

show flow cache

```

Ingress IPv4 Entries
SIP      DIP      BD ID    S-Port  D-Port  Protocol  Byte Count  Packet Count
TCP FLAGS  TOS      if_id    flowStart  flowEnd  Profile   Ing-VRF
17.1.1.2   17.1.1.1  1671     0        0        89        480         8
0xc0      0xc0      0x1a004400 2938966  2976728  5 : NF    0
17.1.1.2   224.0.0.13 1672     0        0        103       144         2
0xc0      0xc0      0x1a004400 2941719  2969951  5 : NF    0
17.1.1.2   224.0.0.13 1675     0        0        103       72          1
0xc0      0xc0      0x1a004400 2961417  2961667  5 : NF    0
17.1.1.2   224.0.0.5  1675     0        0        89        340         5

```



```

      0x0      0xc0      0x1a004400 2943341      2979400      5 : NF      0
17.1.1.2      17.1.1.1      1671      2048      0      1      3612      43
      0x0      0x0      0x1a004400 2938188      2980184      5 : NF      0

Ingress IPV6 Entries
SIP
Count TCP FLAGS Flow Label if_id flowStart flowEnd Ing-VRF
fe80::822d:bfff:fe81:e415 ff02::5 4147 0 0 89 490 5
      0x0      0x0      0x1a003400 11217548 11254367 1

```




Configuring sFlow

This chapter describes how to configure sFlow on Cisco NX-OS devices.

This chapter includes the following sections:

- [sFlow, on page 453](#)
- [Prerequisites for sFlow, on page 454](#)
- [Guidelines and Limitations for sFlow, on page 454](#)
- [Default Settings for sFlow, on page 456](#)
- [Configuring sFlow , on page 457](#)
- [Verifying the sFlow Configuration, on page 465](#)
- [Monitoring and Clearing sFlow Statistics, on page 465](#)
- [Configuration Examples for sFlow, on page 465](#)
- [Additional References, on page 466](#)

sFlow

Sampled flow (sFlow) is a sampled flow monitoring technology that:

- enables you to monitor real-time traffic in data networks with switches and routers,
- employs the sampling mechanism in sFlow agent software to monitor traffic, and
- forwards sample data to a central data collector.

For more information about sFlow, see [RFC 3176](#).

sFlow Agent

The sFlow agent, which is embedded in the Cisco NX-OS software, periodically samples or polls the interface counters that are associated with a data source of the sampled packets. The data source can be an Ethernet interface, an EtherChannel interface, or a range of Ethernet interfaces. The sFlow agent queries the Ethernet port manager for the respective EtherChannel membership information and also receives notifications from the Ethernet port manager for membership changes.

When you enable sFlow sampling, based on the sampling rate and the hardware internal random number, the ingress packets and egress packets are sent to the CPU as an sFlow-sampled packet. The sFlow agent processes the sampled packets and sends an sFlow datagram to the sFlow analyzer. In addition to the original sampled

packet, an sFlow datagram includes information about the ingress port, the egress port, and the original packet length. An sFlow datagram can have multiple sFlow samples.

Prerequisites for sFlow

sFlow has the following prerequisites:

- For Cisco Nexus 9332PQ, 9372PX, 9372TX, and 93120TX switches and for Cisco Nexus 9396PX, 9396TX, and 93128TX switches with the N9K-M6PQ generic expansion module (GEM), you must configure the sFlow and SPAN ACL TCAM region sizes for any uplink ports that are to be configured as an sFlow data source. To do so, use the **hardware access-list tcam region sflow** and **hardware access-list tcam region span** commands. See [Configuring ACL TCAM Region Sizes](#) for more information.



Note By default, the sflow region size is zero, and the span region size is non-zero. You need to configure the sflow region to 256 and allocate enough entries to the span region in order to configure the port as an sFlow data source.

- Egress sFlow of multicast traffic requires **hardware multicast global-tx-span** configuration

Guidelines and Limitations for sFlow



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

sFlow has the following guidelines and limitations:

- If at least one sFlow data source is configured, the SPAN sessions cannot be brought up.
 - If at least one SPAN session is configured as **no shut**, sFlow data sources cannot be added.
- The sampling mode that is used for sFlow is based on an algorithm that is known as LFSR. Due to the use of LFSR, it is not guaranteed that one in every few packets are sampled with the sampling rate of n. However, the number of packets that are sampled is equal to the total packets over a period of time.
- When sFlow is used to sample the Rx traffic from FEX HIF ports, additional VNTAG and 802.1q tags are present in the sampled traffic.
- In Cisco Nexus 9300-EX and 9300-FX platform switches, the FEX, HIF, and NIF ports cannot be configured as sFlow data-source interfaces.
- When sFlow and SPAN are configured on the same interface, and the hardware rate-limiter is configured for sFlow, the Rate-Limiter Drops counter in the output of the **show hardware rate-limiter** command displays more drops than expected.

- sFlow is a software-driven feature, hardware only sends copies of traffic from the sFlow source interfaces to the CPU for further processing. Elevated CPU usage is expected. sFlow traffic sent to the CPU by hardware is rate-limited to protect the CPU.

- When you enable sFlow for an interface, it is enabled for both ingress and egress. You cannot enable sFlow for only ingress or only egress.

For Cisco Nexus 9508 switches with Cisco Nexus 9636C-R and 9636Q-R line cards, sFlow can be enabled for an interface only in the ingress direction.

- The storm control feature does not work if you enable storm control on an interface where sFlow is also enabled.
- sFlow is not supported on the SVIs.
- Subinterfaces are not supported for sFlow.
- We recommend you configure the sampling rate that is based on the sFlow configuration and traffic in the system.
- The switch supports only one sFlow collector.
- sFlow and Network Address Translation (NAT) are not supported on the same port.
- sFlow supports sampling IPv6 traffic.
- sFlow does not support egress sampling for multicast, broadcast, or unknown unicast packets.
- sFlow counters increment even for control packets that ingress on the sFlow data-source interfaces. These packets may be sampled and send out as sFlow datagrams (similar to data plane traffic).
- The following Cisco Nexus switches support sFlow and SPAN together:
 - N9336C-FX2
 - N93240YC-FX2
 - N93360YC-FX2
- Beginning with Cisco NX-OS Release 9.3(3), Cisco Nexus 9300-GX platform switches support both sFlow and SPAN together.
- Nexus 9000-EX, FX, GX family of switches only support sampling at the following values: 4096, 8192, 16384, 32768, 65536. Configuring values other than these results in the value being rounded off to the next supported value.
- When sFlow is configured on N9K-C9508-FM-G with the N9K-X9716D-GX line card, disable sFlow before configuring SPAN sessions.
- Beginning with Cisco NX-OS Release 10.1(1), sFlow and SPAN are supported on Cisco N9K-C93180YC-FX3 platform switches.
- Beginning with Cisco NX-OS Release 10.1(2), sFlow is supported on the Cisco N9K-X9624D-R2 line card.
- Beginning with Cisco NX-OS Release 10.1(2), sFlow supports VXLAN traffic on the Cisco N9K-C9508-FM-G cloud-scale fabric module with the N9K-X9716D-GX line card.

- Beginning with Cisco NX-OS Release 10.2(1), sFlow Extended BGP (Gateway) is supported on the Cisco N9K-C93600CD-GX, N9K-C93240YC-FX2, N9K-C93180YC-EX, N9K-C93180YC-FX, N9K-C93180YC-FX3S, N9K-93600CD-GX platform switches, and Nexus 9500 with N9K-X9716D-GX and N9K-X9736C-FX line cards.
- NX-OS provides flexible forwarding templates to utilize the hardware resources according to customer needs. For sFlow ingress IPv6 sampling to fill BGP information correctly in the sFlow record, a template which has all IPv6 routes on the line-card has to be selected. For example, customers can configure **system routing template-mpls-heavy**. For more information, please refer to the Cisco Nexus 9000 Series NX-OS Command Reference (Configuration Commands), Release 9.3(x). For command to take effect, system needs to be rebooted. This is applicable on GX modular chassis.
- When ECMP is configured in BGP and in case of ECMP destination routes, the next-hop information in the extended gateway record of the exported sFlow record will be 0. Other BGP information like Autonomous System will be derived from the first path. The output interface in the sFlow record will be set to 0 (unknown) to indicate that the flow could be through any of the paths.
- Beginning with Cisco NX-OS Release 10.2(1q)F, sFlow is supported on the Cisco N9K-C9332D-GX2B platform switches.
- Beginning with Cisco NX-OS Release 10.2(1), extended BGP data can now be collected. In order for sFlow to collect this data, a non-SVI Layer 3 interface such as a physical interface or port-channel must be configured as the sFlow source.
- Beginning with Cisco NX-OS Release 10.2(3)F, sFlow flow-cache size is increased from 3k route entries in earlier releases to 30k v4 and 30k v6 route entries. This feature is supported on Cisco Nexus C93600CD-GX, C93240YC-FX2, C93180YC-EX, C93180YC-FX, C93180YC-FX3S, 93600CD-GX, and X9716D-GX platform switches.
- Beginning with Cisco NX-OS Release 10.3(1)F, sFlow is supported on the Cisco Nexus 9808 platform switches.
 - For egress sampled packet, re-written information is not available in sFlow record.
 - Egress sFlow is not supported for directly connected host.
 - sFlow is not supported on the sub-interface traffic.
- Beginning with Cisco NX-OS Release 10.3(1)F, sFlow supports IPv6 collector. However, at a time, only one collector can be configured, either IPv4 or IPv6. Also, the source ip address and the collector ip address must belong to the same address family, that is, either IPv4 or IPv6 address family.

Default Settings for sFlow

The following table lists the default settings for sFlow parameters.

Table 22: Default sFlow Parameters

Parameters	Default
sFlow sampling rate	4096
sFlow sampling size	128

Parameters	Default
sFlow counter poll interval	20
sFlow maximum datagram size	1400
sFlow collector IP address	0.0.0.0
sFlow collector port	6343
sFlow agent IP address	0.0.0.0

Configuring sFlow

Enabling sFlow

You must enable the sFlow feature before you can configure sFlow settings on the switch.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] feature sflow Example: switch(config)# feature sflow	Enables or disables sFlow.
Step 3	(Optional) show feature Example: switch(config)# show feature	Displays the enabled and disabled features.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the Sampling Rate

You can configure the sampling rate for sFlow.

Before you begin

Make sure that you have enabled sFlow.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] sflow sampling-rate <i>sampling-rate</i> Example: <pre>switch(config)# sflow sampling-rate 50000</pre>	Configures the sFlow sampling rate for packets. The <i>sampling-rate</i> can be an integer between 4096 and 1000000000. Note Nexus 9000-EX, FX, GX, FX2, FX3, GX2 family of switches support sampling only at the following values: 4096, 8192, 16384, 32768, 65536. Configuring values other than these will result in the value being rounded off to the next supported value.
Step 3	(Optional) show sflow Example: <pre>switch(config)# show sflow</pre>	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring the Maximum Sampled Size

You can configure the maximum number of bytes that should be copied from a sampled packet.

Before you begin

Make sure that you have enabled sFlow.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] sflow max-sampled-size <i>sampling-size</i>	Configures the sFlow maximum sampling size.

	Command or Action	Purpose
	Example: <pre>switch(config)# sflow max-sampled-size 200</pre>	The range for the <i>sampling-size</i> is from 64 to 256 bytes.
Step 3	(Optional) show sflow Example: <pre>switch(config)# show sflow</pre>	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring the Counter Poll Interval

You can configure the maximum number of seconds between successive samples of the counters that are associated with the data source. A sampling interval of 0 disables counter sampling.

Before you begin

Make sure that you have enabled sFlow.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] sflow counter-poll-interval poll-interval Example: <pre>switch(config)# sflow counter-poll-interval 100</pre>	Configures the sFlow poll interval for an interface. The range for the <i>poll-interval</i> is from 0 to 2147483647 seconds.
Step 3	(Optional) show sflow Example: <pre>switch(config)# show sflow</pre>	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring the Maximum Datagram Size

You can configure the maximum number of data bytes that can be sent in a single sample datagram.

Before you begin

Make sure that you have enabled sFlow.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] sflow max-datagram-size <i>datagram-size</i> Example: switch(config)# sflow max-datagram-size 2000	Configures the sFlow maximum datagram size. The range for the <i>datagram-size</i> is from 200 to 9000 bytes.
Step 3	(Optional) show sflow Example: switch(config)# show sflow	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the sFlow Collector Address

You can configure the IPv4 or IPv6 address of the sFlow data collector that is connected to the management port.

Before you begin

Make sure that you have enabled sFlow.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.

	Command or Action	Purpose
Step 2	[no] sflow collector-ip <i>ip-address</i> vrf <i>vrf</i> [source <i>ip-address</i>] Example: <pre>switch(config)# sflow collector-ip 192.0.2.5 vrf management switch(config)# sflow collector-ip 2001::1 vrf management</pre>	Configures the IPv4 or IPv6 address for the sFlow collector. If the IP address is set to 0.0.0.0, all samples will be dropped. The <i>vrf</i> can be one of the following: • A user-defined VRF name—You can specify a maximum of 32 alphanumeric characters. • vrf management—You must use this option if the sFlow data collector is on the network connected to the management port. • vrf default—You must use this option if the sFlow data collector is on the network connected to the front-panel ports. The source <i>ip-address</i> option causes the sent sFlow datagram to use the source IP address as the IP packet source address. The source IP address has to be already configured on one of the switch local interfaces; otherwise, an error message appears. If the interface with the source IP address is changed or removed after this option is configured, the sFlow datagram will no longer be sent out, and an event history error and syslog error will be logged. When the source <i>ip-address</i> option is not configured, Cisco NX-OS picks the IP packet source address automatically for the sent sFlow datagram.
Step 3	(Optional) show sflow Example: <pre>switch(config)# show sflow</pre>	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring the sFlow Collector Port

You can configure the destination port for sFlow datagrams.

Before you begin

Make sure that you have enabled sFlow.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] sflow collector-port collector-port Example: <pre>switch(config)# sflow collector-port 7000</pre>	Configures the UDP port of the sFlow collector. The range for the <i>collector-port</i> is from 1 to 65535.
Step 3	(Optional) show sflow Example: <pre>switch(config)# show sflow</pre>	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring the sFlow Agent Address

You can configure the IPv4 or IPv6 address of the sFlow agent.

Before you begin

Make sure that you have enabled sFlow.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no] sflow agent-ip ip-address Example: <pre>switch(config)# sflow agent-ip 192.0.2.3 switch(config)# sflow agent-ip 2001::10</pre>	Configures the IPv4 or IPv6 address of the sFlow agent. The default IP address is 0.0.0.0, which means that all samples will be dropped. You must specify a valid IP address to enable sFlow functionality. Note

	Command or Action	Purpose
		This IP address is not necessarily the source IP address for sending the sFlow datagram to the collector. The agent ip address and the collector ip address must belong to the same address family, that is, either IPv4 or IPv6 address family.
Step 3	(Optional) show sflow Example: <code>switch(config)# show sflow</code>	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Configuring the sFlow Sampling Data Source

You can configure the source of the data for the sFlow sampler as an Ethernet port, a range of Ethernet ports, or a port channel.

Before you begin

Make sure that you have enabled sFlow.

If you want to use a port channel as the data source, make sure that you have already configured the port channel and you know the port channel number.

Make sure that the sFlow and SPAN ACL TCAM region sizes are configured for any uplink ports that are to be configured as an sFlow data source on the following devices: Cisco Nexus 9332PQ, 9372PX, 9372TX, and 93120TX switches and Cisco Nexus 9396PX, 9396TX, and 93128TX switches with the N9K-M6PQ generic expansion module (GEM).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	[no] sflow data-source interface [ethernet slot/port[-port] port-channel channel-number] Example:	Configures the sFlow sampling data source. For an Ethernet data source, <i>slot</i> is the slot number, and <i>port</i> can be either a single port

	Command or Action	Purpose
	<code>switch(config)# sflow data-source interface ethernet 1/5-12</code>	number or a range of ports designated as <i>port-port</i> .
Step 3	(Optional) show sflow Example: <code>switch(config)# show sflow</code>	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Configuring sFlow Extended BGP (Gateway)

You can configure sFlow Extended BGP on the switch.

Before you begin

Make sure that you have enabled sFlow.

Make sure that the source port is a non-SVI Layer 3 interface, such as a physical interface or port-channel.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal switch(config)#</code>	Enters global configuration mode.
Step 2	[no] sflow extended bgp Example: <code>switch(config)# sflow extended bgp</code>	Configures extended bgp on the switch. Sampled sFlow packets with destination IP address to BGP installed routes will include extended gateway (bgp) data in the exported sFlow record.
Step 3	(Optional) show sflow Example: <code>switch(config)# show sflow</code>	Displays the sFlow configuration.
Step 4	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Verifying the sFlow Configuration

Use these commands to display the sFlow configuration.

Table 23: sFlow Show Commands

Command	Purpose
show sflow	Displays all the data sources of the sFlow samplers and the sFlow agent configuration.
show process	Verifies whether the sFlow process is running.
show running-config sflow [all]	Displays the current sFlow running configuration.

Monitoring and Clearing sFlow Statistics

Use the **show sflow statistics** command to display the sFlow statistics.

Use the following commands to clear the sFlow statistics:

Command	Description
clear sflow statistics	Clears most of the sFlow statistics from the show sflow statistics command.
clear counters interface all	Clears the Total Packets field from the show sflow statistics command.
clear hardware rate-limiter sflow	Clears the Total Samples field from the show sflow statistics command.

Configuration Examples for sFlow

This example shows how to configure sFlow:

```
feature sflow
sflow sampling-rate 4096
sflow max-sampled-size 200
sflow counter-poll-interval 100
sflow max-datagram-size 2000
sflow collector-port 7000
sflow agent-ip 192.0.2.3
sflow collector-ip 192.0.2.5 vrf management
sflow data-source interface ethernet 1/5
```

Additional References

Related Documents

Related Topic	Document Title
ACL TCAM regions	Configuring IP ACLs



CHAPTER 28

Configuring TAP Aggregation and MPLS Stripping

This chapter describes how to configure TAP aggregation and MPLS stripping on Cisco NX-OS devices.

This chapter contains the following sections:

- [About TAP Aggregation, on page 467](#)
- [About MPLS Stripping, on page 470](#)
- [Configuring TAP Aggregation, on page 472](#)
- [Verifying the TAP Aggregation Configuration, on page 476](#)
- [Configuration Example for TAP Aggregation, on page 477](#)
- [Configuring MPLS Stripping, on page 477](#)
- [Clearing MPLS Stripping Counters and Label Entries, on page 481](#)
- [Configuration Examples for MPLS Stripping, on page 482](#)
- [Additional References, on page 482](#)

About TAP Aggregation

Network TAPs

You can use various methods to monitor packets. One method uses physical hardware test access points (TAPs).

Network TAPs can be extremely useful in monitoring traffic because they provide direct inline access to data that flows through the network. In many cases, a third party monitors the traffic between two points in the network. If the network between points A and B consists of a physical cable, a network TAP might be the best way to accomplish this monitoring. The network TAP has at least three ports: an A port, a B port, and a monitor port. A TAP inserted between the A and B ports passes all traffic through unimpeded, but it also copies that same data to its monitor port, which could enable a third party to listen.

TAPs have the following benefits:

- They can handle full-duplex data transmission.
- They are unobtrusive and not detectable by the network (with no physical or logical addressing).
- Some TAPs support full inline power with the capability to build a distributed TAP.

If you are trying to gain visibility into the server-to-server data communication at the edge or virtual edge of your network or to provide a copy of traffic to the Intrusion Prevention System (IPS) appliance at the Internet edge of your network, you can use network TAPs nearly anywhere in the environment. However, this deployment can add significant costs, operation complexities, and cabling challenges in a large-scale environment.

TAP Aggregation

TAP aggregation is an alternative solution to help with monitoring and troubleshooting tasks in the data center. It works by designating a device to allow the aggregation of multiple test access points (TAPs) and to connect to multiple monitoring systems. TAP aggregation switches link all of the monitoring devices to specific points in the network fabric that handle the packets that need to be observed.

In the TAP aggregation switch solution, a Cisco Nexus 9000 Series switch is connected to various points in the network at which packet monitoring is advantageous. From each network element, you can use switched port analyzer (SPAN) ports or optical TAPs to send traffic flows directly to this TAP aggregation switch. The TAP aggregation switch is directly connected to all of the analysis tools used to monitor the events in the network fabric. These monitoring devices include remote monitor (RMON) probes, application firewalls, IPS devices, and packet sniffer tools.

You can configure the TAP aggregation switch to filter specific traffic and redirect it to one or more tools. In order to redirect the traffic to multiple interfaces, a multicast group is created internally on the switch, and the interfaces that are part of the redirect list are added as member ports. When an access control list (ACL) policy with the redirect action is applied to an interface, the traffic matching the ACL rule is redirected to the internal multicast group that is created.

Guidelines and Limitations for TAP Aggregation



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

TAP aggregation has the following guidelines and limitations:

- TAP aggregation:
 - Supported on all Cisco Nexus 9000 Series switches and the 3164Q, 31128PQ, 3232C, and 3264Q switches.
 - Supported on 100G ports.
 - Supports only on switch ports and only in the ingress direction.
 - Supports IPv4 ACLs with UDF-based match for Cisco Nexus 9200, 9300, and 9300-EX Series switches.
 - Supported on Cisco Nexus 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, 9300-GX2, 9500-EX, and 9500-FX platform switches.
 - Maximum redirect ports supported are 32 interfaces.
- Beginning with Cisco NX-OS Release 9.2(1), TAP aggregation filters on MPLS tags are supported on the following Cisco Nexus platform switches:

- Cisco Nexus 9000 platform switches, including the 9700-EX and 9700-FX line cards.
 - Cisco Nexus 9200 platform switches.
 - Cisco Nexus 9300 platform switches.
 - Cisco Nexus 9500 switches.
- TAP aggregation filters on MPLS tags are not supported on the following Cisco Nexus Series switches, line cards, and fabric modules:

Table 24: Cisco Nexus 9000 Series Switches

Cisco Nexus 3164Q-40GE	Cisco Nexus 9372PX	Cisco Nexus 9372PX-E
Cisco Nexus 9372TX	Cisco Nexus 9372TX-E	Cisco Nexus 9332PQ
Cisco Nexus 3232C	Cisco Nexus 93120TX	Cisco Nexus 31128PQ
Cisco Nexus 3264Q-S	—	—

Table 25: Cisco Nexus 9000 Series Line Cards and Fabric Modules

N9K-M6PQ	N9K-X9632PC-QSFP100	N9K-X9536PQ
N9K-X9432C-S	N9K-C93128TX	N9K-C9396PX
N9K-X9432PQ	N9K-X9464TX	—

- Cisco Nexus 9700-EX and 9700-FX line cards support TAP aggregation with IPv4, IPv6, and MAC ACLs.
- Only Layer 2 interfaces support the TAP aggregation policy. You can apply the policy to a Layer 3 interface, but the policy becomes nonfunctional.
- The redirect port must be part of the same VLAN as the source (TAP) port.
- Each rule must be associated with only one unique match criterion.
- When you enter a list of interfaces for the TAP aggregation policy, you must separate them with commas but no spaces. For example, port-channel50, ethernet1/12, port-channel20.
- When you specify target interfaces in a policy, make sure that you enter the whole interface type and not an abbreviated version. For example, make sure that you enter **ethernet1/1** instead of **eth1/1** and **port-channel50** instead of **po50**.
- HTTP requests with *tcp-option-length* and *VLAN ID* filters simultaneously are not supported. Traffic match against ACE may not work if you configure both filters at a time.
- Beginning with Cisco NX-OS Release 10.2(1)F, the TAP aggregation feature is licensed and requires you to configure feature tap-aggregation before configuring related CLIs. However, this feature is auto-generated during ISSU infra-convert phase of sysmgr if any tap-aggregation dependent CLI usage is found in the earlier configurations. This feature is supported on all Cisco Nexus 9000 Series switches. For more information about licensing, refer to *Cisco Nexus 9000 NX-OS Smart Licensing Using Policy Guide*.

- Beginning with Cisco NX-OS Release 10.2(2)F, ensure that you configure the **mode tap-aggregation** command before attaching TapAgg ACLs on L2 interface.
- When configuring ACL entries with redirect to port-channels that are yet to be configured, the user must take care to configure the specified port-channels at a later point of time.
- Beginning with Cisco NX-OS Release 10.3(1)F, on the interface with selective Q-in-Q trunk mode the provider VLAN tagging is supported on the Cisco Nexus 9300-GX, N9K-C9504-FM-G, and N9K-C9508-FM-G switches and N9K-X9716D-GX line cards with below limitations:
 - If VXLAN is enabled, this feature is not supported.
 - You can have maximum 7000 outer VXLAN late entries at overall system level and 4000 entries per port.
- To allow double VLAN tags on ingress interface, the **switchport trunk allow-multi-tag** command must be configured correctly as mentioned below:
 - On Cisco Nexus 9300-FX2 switches, this command must be used only if NDB is configured.
 - On Cisco Nexus 9300-GX/GX2 switches, this command is not required if NDB is configured.
- A few guidelines and limitations for the **hardware acl tap-agg** command include:
 - This command is required on Cisco Nexus 9300-GX platform switches for the header stripping functionality to work. After the configuration a switch reload is required.
 - If **mode tap-aggregation** is configured under interfaces on Cisco Nexus 9300-GX switches, this command is mandatory.
 - This command is not required on Cisco Nexus 9300-FX3 platforms.

About MPLS Stripping

The ingress ports of Cisco Nexus 9000 Series switches receive various Multiprotocol Label Switching (MPLS) packet types. Each data packet in an MPLS network has one or more label headers. These packets are redirected on the basis of a redirect access control list (ACL).

A label is a short, four-byte, fixed-length, locally significant identifier that is used to identify a Forwarding Equivalence Class (FEC). The label that is put on a particular packet represents the FEC to which that packet is assigned. It has the following components:

- Label—Label value (unstructured), 20 bits
- Exp—Experimental use, 3 bits; currently used as a class of service (CoS) field
- S—Bottom of stack, 1 bit
- TTL—Time to live, 8 bits

Standard network monitoring devices cannot monitor and analyze the MPLS traffic. You need to enable the MPLS strip feature to allow the standard network monitoring tools to monitor the MPLS traffic. This feature strips off the MPLS label headers of the traffic and redirects the traffic to the monitoring devices.

Guidelines and Limitations for MPLS Stripping



Note For scale information, see the release-specific *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

MPLS stripping has the following guidelines and limitations:

- Cisco Nexus 9700-EX and 9700-FX line cards do not support MPLS stripping.
- Beginning from Cisco NX-OS Release 10.2(1)F, **feature tap-aggregation** must be enabled for all Tap Aggregation and stripping functions.
- Disable all Layer 3 and vPC features before you enable MPLS stripping.
- Static MPLS, MPLS segment routing, and MPLS stripping cannot be enabled at the same time.
- Only the ingress interfaces involved in MPLS stripping must have TAP aggregation enabled.
- You must configure the TAP aggregation ACL with a redirect action on the ingress interface to forward the packet to the desired destination.
- Post MPLS strip, SMAC changes to switch mac (**show vdc**) and DMAC is set to **00:00:00:ab:cd:ef**.
- The egress interface where stripped packets will exit must be an interface that has VLAN 1 as an allowed VLAN. We recommend that you configure the egress interface as a trunk with all VLANs allowed by default.
- Stripping is based on IP PACL, and you cannot use MAC-ACL for stripping.
- MPLS stripping is supported only for IPv4 traffic.
- Port-channel load balancing is supported for MPLS stripped packets.
- Layer 3 header-based hashing and Layer 4 header-based hashing are supported, but Layer 2 header-based hashing is not supported.
- During MPLS stripping, the incoming VLAN is not preserved.
- Cisco Nexus 9200, 9300-EX, and 9300-FX platform switches support tagging of VLANs to packets going out of redirect ports. The ingress/egress ports can either be ethernet or port channel. The VLAN tag is derived from the incoming port configuration. The new ACL on the ingress interface should not be associated with a VLAN value different from the interface VLAN value.
- For every ACE (under an ACL associated with a particular VLAN) with a unique redirect port list, we allocate a hardware entry. The current hardware limit for the number of ACEs is 50 and you cannot configure more than 50 such ACEs.
- MPLS strip is only supported for Layer 3 packets under the MPLS label stack.
- Beginning with Cisco NX-OS Release 10.2(2)F, EoMPLS label stripping is supported only on Cisco Nexus 9300-EX platform switches. However, VPLS strip and control-word packet strip is not supported.
- Beginning with Cisco NX-OS Release 10.2(3)F, OFM-based MPLS stripping is added. The new OFM-based MPLS stripping and legacy implementation cannot co-exist. For more information, see the OFM-based MPLS header strip section under [Configuring Header Stripping Features for Nexus Data Broker](#), on page 487.

- Use the new OFM-based MPLS stripping feature only if the deployment needs co-existence of MPLS stripping with any other type of header stripping such as VXLAN, iVXLAN, GRE, and ERSPAN headers. The existing MPLS stripping feature will continue to support MPLS stripping when co-existence is not needed with other stripping features.
- Beginning with Cisco NX-OS Release 10.3(2)F, EoMPLS label stripping is also supported on Cisco Nexus 9300-FX ToR switches.

Configuring TAP Aggregation

Enabling TAP Aggregation for Line Cards

Beginning with Cisco NX-OS Release 7.0(3)I7(2), you can enable TAP aggregation for Cisco Nexus 9500 platform switches with 9700-EX and 9700-FX line cards.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code>	Enters global configuration mode.
Step 2	[no] hardware acl tap-agg Example: <code>switch(config)# hardware acl tap-agg</code>	Enables TAP aggregation for Cisco Nexus 9700-EX and 9700-FX line cards. This command is also needed on Cisco Nexus 9300-GX and 9300-GX2 platform switches and may require reload.
Step 3	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.

Configuring a TAP Aggregation Policy

You can configure a TAP aggregation policy on an IP access control list (ACL) or on a MAC ACL.

Before you begin

You must configure the ACL TCAM region size for IPv4 port ACLs or MAC port ACLs using the **hardware access-list tcam region** *{ifacl | mac-ifacl}* command. Configure the ACL TCAM region size for IPv6 port ACLs using the command, **hardware access-list team region ipv6-ifcal**.

For information, see the "Configuring ACL TCAM Region Sizes" in the [Cisco Nexus 9000 Series NX-OS Security Configuration Guide](#).



Note By default the region size for both ifacl and mac-ifacl is zero. You need to allocate enough entries to the ifacl or mac-ifacl region to support TAP aggregation.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	feature tap-aggregation Example: <pre>switch(config)# feature tap-aggregation switch(config)#</pre>	Allows you to configure to CLIs related to tap-aggregation. Note Beginning with Cisco NX-OS Release 10.2(1)F, for software upgrades from earlier releases to the newer NX-OS release with this feature, if ISSU is completed on a supported matrix, the feature tap-aggregation configuration is automatically generated.
Step 3	Enter one of the following commands: • ip access-list <i>access-list-name</i> • mac access-list <i>access-list-name</i> Example: <pre>switch(config)# ip access-list test switch(config-acl)# switch(config)# mac access-list mactap1 switch(config-mac-acl)#</pre>	Creates an IPACL and enters IP access list configuration mode or creates a MAC ACL and enters MAC access list configuration mode.
Step 4	(Optional) statistics per-entry Example: <pre>switch(config-acl)# statistics per-entry</pre>	Starts recording statistics for how many packets are permitted or denied by each entry.
Step 5	[no] permit protocol source destination redirect interfaces Example: <pre>switch(config-acl)# permit ip any any redirect ethernet1/8</pre>	Creates an IP or MAC ACL rule that permits traffic to be redirected per its conditions. The no version of this command removes the permit rule from the policy. Note When you enter an interface for the TAP aggregation policy, do not abbreviate it. When you enter a list of interfaces, separate them with commas but no spaces.

	Command or Action	Purpose
Step 6	(Optional) Enter one of the following commands: • show ip access-lists <i>[access-list-name]</i> • show mac access-lists <i>[access-list-name]</i> Example: <pre>switch(config-acl)# show ip access-lists test</pre> <pre>switch(config-mac-acl)# show mac access-lists mactapl</pre>	Displays all IPv4 or MAC ACLs or a specific IPv4 or MAC ACL.
Step 7	(Optional) copy running-config startup-config Example: <pre>switch(config-acl)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Attaching a TAP Aggregation Policy to an Interface

You can apply an ACL configured with TAP aggregation to a Layer 2 interface.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal</pre> <pre>switch(config)#</pre>	Enters global configuration mode.
Step 2	interface <i>type slot/port</i> Example: <pre>switch(config)# interface ethernet 2/2</pre> <pre>switch(config-if)#</pre>	Enters interface configuration mode for the specified interface.
Step 3	switchport Example: <pre>switch(config-if)# switchport</pre>	Changes a Layer 3 interface to a Layer 2 interface. Note Make sure that the interface is a Layer 2 interface.
Step 4	Enter one of the following commands: • [no] ip port access-group <i>access-list-name</i> in • [no] mac port access-group <i>access-list-name</i> in	Applies an IPv4 or MAC ACL configured with TAP aggregation to the interface. The no form of this command removes the ACL from the interface.

	Command or Action	Purpose
	Example: <pre>switch(config-if)# ip port access-group test in</pre> <pre>switch(config-if)# mac port access-group test in</pre>	
Step 5	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring Selective Q-in-Q with Provider VLANs

Before you begin

You must configure provider VLANs

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal</pre>	Enters global configuration mode.
Step 2	interface interface-id Example: <pre>switch(config)# interface Ethernet1/1</pre>	Enters interface configuration mode for the interface connected to the service provider network. You can enter a physical interface or an EtherChannel port channel.
Step 3	switchport Example: <pre>switch(config-if)# switchport</pre>	Sets the interface as a Layer 2 switching port.
Step 4	switchport mode trunk Example: <pre>switch(config-if)# switchport mode trunk</pre>	Sets the interface as a Layer 2 trunk port.
Step 5	Enter one of the following commands: switchport vlan mapping <i>vlan-id-range</i>dot1q-tunnel <i>outer vlan-id</i> switchport vlan mapping all dot1q-tunnel <i>outer vlan-id</i> Example:	Enter the VLAN IDs to be mapped: vlan-id-range—The customer VLAN ID range(C-VLAN) entering the switch from the customer network. The range is from 1 to 4094. You can enter a string of VLAN-IDs.

	Command or Action	Purpose
	<code>switch(config-if)# switchport vlan mapping all dot1q-tunnel 300</code>	outer vlan-id—Enter the outer VLAN ID (S-VLAN) of the service provider network. The range is from 1 to 4094.
Step 6	switchport trunk allowed vlan <i>vlan_list</i> Example: <code>switch(config-if)# switchport trunk allowed vlan 300</code>	Sets the allowed VLANs for the trunk interface.
Step 7	Enter one of the following commands: [no] ip port access-group <i>access-list-name</i> in [no] mac port access-group <i>access-list-name</i> in Example: <code>switch(config-if)# ip port access-group test in</code> <code>switch(config-if)# mac port access-group test in</code>	Applies an IPv4 or MAC ACL configured with TAP aggregation to the interface. The no form of this command removes the ACL from the interface.
Step 8	(Optional) mode tap-aggregation Example: <code>switch(config-if)# mode tap-aggregation</code> <code>switch(config-if)# no shutdown</code>	Allows the attachment of an ACL with the tap aggregation policy to the interface.
Step 9	(Optional) copy running-config startup-config Example: <code>switch(config-if)# copy running-config startup-config</code>	Copies the running configuration to the startup configuration.
Step 10	<code>switch(config-if)# exit</code>	Exits the configuration mode.
Step 11	(Optional) <code>switch(config-if)# show interfaces <i>interface-id</i> vlan mapping</code>	Verifies the mapping configuration.

Verifying the TAP Aggregation Configuration

To display the TAP aggregation configuration information, perform one of the following tasks.

Command	Purpose
<code>show ip access-lists [<i>access-list-name</i>]</code>	Displays all IPv4 ACLs or a specific IPv4 ACL.
<code>show mac access-lists [<i>access-list-name</i>]</code>	Displays all MAC ACLs or a specific MAC ACL.

Configuration Example for TAP Aggregation

This example shows how to configure a TAP aggregation policy on an IPv4 ACL:

```
switch# configure terminal
switch(config)# feature tap-aggregation
switch(config)# ip access-list test
switch(config-acl)# 10 deny ip 100.1.1/24 any
switch(config-acl)# 20 permit tcp any eq www any redirect port-channel4
switch(config-acl)# 30 permit ip any any redirect
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
switch(config-acl)# show ip access-lists test
IP access list test
    10 deny ip 100.1.1/24 any
    20 permit tcp any eq www any redirect port-channel4
    30 permit ip any any redirect
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
```

This example shows how to configure a TAP aggregation policy on a MAC ACL:

```
switch# configure terminal
switch(config)# feature tap-aggregation
switch(config)# mac access-list mactap1
switch(config-mac-acl)# 10 permit any any 0x86dd redirect port-channel1
switch(config-mac-acl)# show mac access-lists mactap1
MAC access list mactap1
    10 permit any any 0x86dd redirect port-channel1
```

This example shows how to attach a TAP aggregation policy to a Layer 2 interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# ip port access-group test in
switch(config-if)#
```

Configuring MPLS Stripping

Enabling MPLS Stripping

You can enable MPLS stripping globally.

Before you begin

Disable all Layer 3 and vPC features before you enable MPLS stripping.

Attach an ACL with the tap aggregation policy to the Layer 2 interface or port channel using the **mode tap-aggregation** command. For more information, see [Attaching a TAP Aggregation Policy to an Interface, on page 474](#).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	[no] mpls strip Example: switch(config)# mpls strip	Globally enables MPLS stripping. The no form of this command disables MPLS stripping.
Step 3	[no] mpls strip mode dot1q Example: switch(config)# mpls strip mode dot1q	Enables VLAN tagging on the packets coming from the redirect port. The VLAN that needs to be tagged must be specified in the ingress port.
Step 4	Required: copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring the Incoming Port for the VLAN Tag

The VLAN tag is derived from the incoming port configuration. The ingress/egress ports can either be ethernet or port channel.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	interface type slot/port Example: switch(config)# interface ethernet 1/26 switch(config-if)#	Enters interface configuration mode for the specified interface.
Step 3	switchport Example: switch(config-if)# switchport	Changes a Layer 3 interface to a Layer 2 interface. Note Make sure that the interface is a Layer 2 interface.

	Command or Action	Purpose
Step 4	Enter one of the following commands: • [no] ip port access-group access-list-name in • [no] mac port access-group access-list-name in Example: <pre>switch(config-if)# ip port access-group test in switch(config-if)# mac port access-group test in</pre>	Applies an IPv4 or MAC ACL configured with TAP aggregation to the interface. The no form of this command removes the ACL from the interface.
Step 5	Enter one of the following commands: • [no] ip port access-group access-list-name in • [no] mac port access-group access-list-name in Example: <pre>switch(config-if)# ip port access-group test in switch(config-if)# mac port access-group test in</pre>	Applies an IPv4 or MAC ACL configured with TAP aggregation to the interface. The no form of this command removes the ACL from the interface.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config-if)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Adding and Deleting MPLS Labels

The device can learn the labels dynamically whenever a frame is received with an unknown label on a TAP interface. You can also add or delete static MPLS labels.

Before you begin

Configure a TAP aggregation policy and attach the policy to an interface. For more information, see the *Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*.

You must configure the TAP aggregation ACL with a redirect action on the ingress interface to forward the packet to the desired destination.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	mpls strip label <i>label</i> Example: <pre>switch(config)# mpls strip label 100</pre>	Adds the specified static MPLS label. The 20-bit value of the label can range from 1 to 1048575. Note This CLI is available for all the platform switches specified for the MPLS Stripping feature in the Guidelines and Limitations section, except for the following cloud scale platform switches: • N9K-C93180YC-EX • N9K-C93180YC-FX • N9K-C93240YC-FX2 • N9K-C93180YC-FX3S • N9K-C93600CD-GX The [no] mpls strip label {<i>label</i> all} command deletes the specified static MPLS label. The all option deletes all static MPLS labels.
Step 3	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring Destination MAC Addresses

You can configure the destination MAC address for stripped egress frames.

Procedure

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

	Command or Action	Purpose
	switch# configure terminal switch(config)#	
Step 2	mpls strip dest-mac <i>mac-address</i> Example: switch(config)# mpls strip dest-mac 1.1.1	Specifies the destination MAC address for egress frames that are stripped of their headers. The MAC address can be specified in one of the following four formats: • E.E.E • EE-EE-EE-EE-EE-EE • EE:EE:EE:EE:EE:EE • EEEE.EEEE.EEEE
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Configuring MPLS Label Aging

You can define the amount of time after which dynamic MPLS labels will age out, if unused.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	mpls strip label-age <i>age</i> Example: switch(config)# mpls strip label-age 300	Specifies the amount of time in seconds after which dynamic MPLS labels age out. The range is from 61 to 31622400.
Step 3	(Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Clearing MPLS Stripping Counters and Label Entries

To clear the MPLS stripping counters and label entries, perform these tasks:

Command	Purpose
clear mpls strip label dynamic	Clears dynamic label entries from the MPLS label table.
clear counters mpls strip	Clears all MPLS stripping counters.

The following example shows how to clear all MPLS stripping counters:

```
switch# clear counters mpls strip
switch# show mpls strip labels
MPLS Strip Labels:
  Total      : 15000
  Static     : 2
Legend:      * - Static Label
  Interface - where label was first learned
  Idle-Age  - Seconds since last use
  SW-Counter- Packets received in Software
  HW-Counter- Packets switched in Hardware
```

Label	Interface	Idle-Age	SW-Counter	HW-Counter
4096	Eth1/44	15	0	0
8192	Eth1/44	17	0	0
12288	Eth1/44	15	0	0
16384	Eth1/44	39	0	0
20480	Eth1/44	47	0	0
24576	Eth1/44	7	0	0
28672	Eth1/44	5	0	0
36864	Eth1/44	7	0	0
40960	Eth1/44	19	0	0
45056	Eth1/44	9	0	0
49152	Eth1/44	45	0	0
53248	Eth1/44	9	0	0

Configuration Examples for MPLS Stripping

This example shows how to add static MPLS labels:

```
switch# configure terminal
switch(config)# mpls strip label 100
switch(config)# mpls strip label 200
switch(config)# mpls strip label 300
```

Additional References

Related Documents

Related Topic	Document Title
IP ACLs	<i>Cisco Nexus 9000 Series NX-OS Security Configuration Guide</i>
MAC ACLs	<i>Cisco Nexus 9000 Series NX-OS Security Configuration Guide</i>

Related Topic	Document Title
Port-channel symmetric hashing	<i>Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide</i>
Remote monitoring (RMON)	Configuring RMON, on page 277
Switched port analyzer (SPAN)	Switched Port Analyzer, on page 351
Troubleshooting	<i>Cisco Nexus 9000 Series NX-OS Troubleshooting Guide</i>



CHAPTER 29

Configuring MPLS Access Lists

- [Configuring MPLS Access Lists, on page 485](#)
- [Verifying the MPLS Access Lists Configuration, on page 486](#)
- [Configuration Examples for MPLS Access Lists, on page 486](#)

Configuring MPLS Access Lists

MPLS Access lists enables filtering of MPLS packets based on MPLS label and sending filtered packets to configured redirect interfaces.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	[no]install feature-set mpls Example: <pre>switch(config)# install feature-set mpls switch(config)# feature-set mpls switch(config)# feature mpls segment-routing</pre>	Enables parsing of MPLS packets. This is mandatory to filter MPLS packets based on MPLS label.
Step 3	mpls access list mpls-acl Example: <pre>switch(config)# mpls access list mpls-acl switch(config-mpls-acl)# 10 permit mpls 1600 any redirect Ethernet1/15</pre>	Configures mpls-access list with filtering based on incoming outer MPLS label. In this example, MPLS packets with incoming label 1600 matched and are redirected to Ethernet1/15.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	(Optional) Copies the running configuration to the startup configuration.

Verifying the MPLS Access Lists Configuration

To display the MPLS access list configuration, perform the following task:

Command	Purpose
show mpls access lists	Displays information about MPLS access lists.

Configuration Examples for MPLS Access Lists

This example shows how to configure MPLS access lists:

```
switch# configure terminal
switch(config)# install feature-set mpls
switch(config)# feature-set mpls
switch(config)# feature mpls segment-routing
switch(config)# mpls access list mpls-acl
switch(config-mpls-acl)# 10 permit mpls 1600 any redirect Ethernet1/15
switch(config)# copy running-config startup-config
```



CHAPTER 30

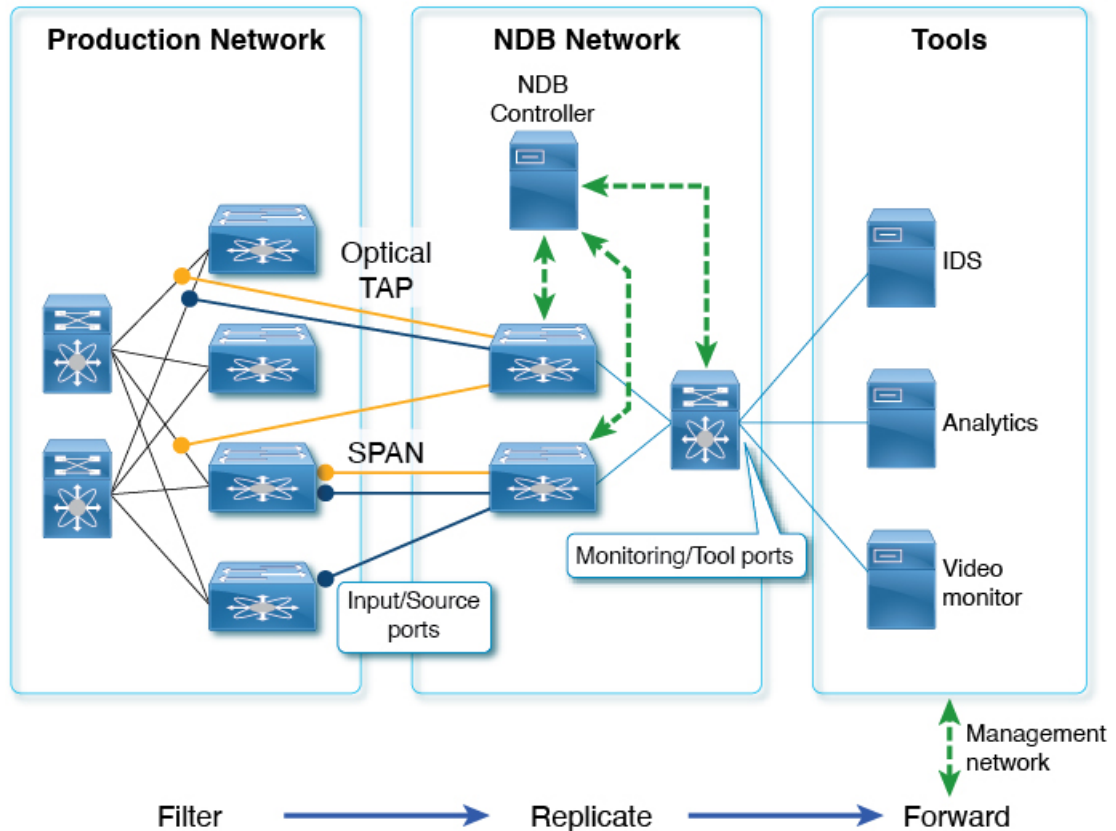
Configuring Header Stripping Features for Nexus Data Broker

- [Introduction to Header Stripping Features for Nexus Data Broker, on page 487](#)
- [Guidelines and Limitations for Header Stripping, on page 489](#)
- [VXLAN and iVXLAN Header Stripping for Nexus Data Broker, on page 490](#)
- [ERSPAN Header Stripping for Nexus Data Broker, on page 494](#)
- [GRE Header Stripping for Nexus Data Broker, on page 498](#)
- [MPLS Header Stripping for Nexus Data Broker, on page 500](#)

Introduction to Header Stripping Features for Nexus Data Broker

Cisco Nexus Data Broker (NDB) builds scalable packet broker network solutions that are easy to operate. The Cisco Nexus Dashboard Data Broker controller software and Cisco Nexus switches provide a new software-defined approach for monitoring both out-of-band and inline network traffic.

Figure 8: NDB Centralized Deployment Model



NDB switches are used for packet monitoring. Packet monitoring is needed for performance monitoring, intrusion detection, check compliance, and so on.

For header strip, Out-of-Band monitoring is done, which means it is non-intrusive, and the copy of the packet is monitored using TAP or SPAN. So, the traffic is filtered and replicated from production network, stripped off any headers on NDB switches, and forwarded to Tools for monitoring. Input/source ports mentioned here are the ports on which the header stripping takes place. Monitoring/Tool ports are the ports which are connected directly to Tools.

The reasons for removing the header are as follows:

- Some monitoring tools do not understand an encapsulated packet.
- Presence of an additional header skews the analytics data.
- Addition of a header adds to the packet size, hampering the optimization of the amount of data that is sent to and processed by the tools.

The benefits of the packet header or label stripping feature of Cisco Nexus Data Broker switch are as follows:

- Enable Multiprotocol Label Switching (MPLS) label stripping
- Native support for VXLAN header stripping from copy traffic
- Support for Generic Route Encapsulation (GRE) header stripping

- Q-in-Q VLAN header stripping at egress

Thus, NDB aligns the legacy VXLAN, IVXLAN, ERSPAN, GRE, and MPLS stripping functionality to the Overlay Forwarding Manager (OFM) based model. The OFM hosts the command line interface (CLI) for header stripping functionality.

This chapter contains the following sections:

- [VXLAN and IVXLAN Header Stripping for Nexus Data Broker](#)
- [ERSPAN Header Stripping for Nexus Data Broker](#)
- [GRE Header Stripping for Nexus Data Broker](#)
- [MPLS Header Stripping for Nexus Data Broker](#)

Guidelines and Limitations for Header Stripping

The guidelines and limitations applicable to all the header stripping features are as follows:

- A maximum of 500 flow terminate interfaces are supported across all tunnel-profiles with various encapsulation types such as VxLAN, iVxLAN, GRE, and MPLS. For ERSPAN, the maximum flow terminate interfaces supported is 31.
- Beginning with Cisco NX-OS Release 10.2(3)F, the MPLS stripping using the OFM model co-exists with the other stripping features. However, the existing MPLS stripping feature will continue to support MPLS stripping when co-existence is not needed with other type of stripping features.
- The co-existence can be on the same interface or different interfaces.



Note

Beginning with Cisco NX-OS Release 10.2(3)F, ERSPAN coexistence on the same interface is supported. However, this is supported on 9300-FX2 and later platforms only.

- The legacy MPLS stripping feature and OFM stripping features are mutually exclusive.
- Beginning with Cisco NX-OS Release 10.2(3)F, traffic with IPv6 inner packet is supported for all stripping functions.
- After performing non-disruptive ISSU from an earlier release to Cisco NX-OS Release 10.2(3)F and performing any header stripping functions, if dot1q tunnel VLAN_tag is missing or set to vlan_id=1, then remove and add the port ACL from L2 interfaces for that particular stripping-enabled interface.
- If no VLAN is configured on an interface, but the switchport mode dot1q-tunnel command is configured on that interface, then stripped packets will have VLAN=1 by default.
- In a scenario where incompatible OFM commands are present in the show running command output, and disruptive ISSU from Cisco NX-OS Release 10.2(3)F to an earlier release is done, wherein OFM commands were not supported in the earlier NX-OS version, then appropriate errors are displayed. However, the show incompatibility command does not flag such errors for OFM-related incompatibility commands.

- The OFM-based GRE, ERSPAN, and MPLS stripping features are supported only on TORs, not on line cards.
- As part of the encapsulation (iVXLAN, VXLAN, GRE, MPLS, ERSPAN), the following restrictions are common:
 - Two or more tunnel-profiles cannot have the same encapsulation-type.
 - OFM-based header stripping features are not supported when feature tunnel is enabled.

VXLAN and iVXLAN Header Stripping for Nexus Data Broker

This subchapter describes VXLAN and iVXLAN header stripping procedure for Nexus Data Broker (NDB).

This chapter contains the following sections:

About Nexus Data Broker – VXLAN and iVXLAN Header Stripping

Nexus Data Broker (NDB) VXLAN, and iVXLAN termination allow switches the ability to strip headers when VXLAN, and iVXLAN packets are received.

NDB switch receives packets in the below mentioned scenarios:

- Test Access Point (TAP) ports between spines and leaf are placed on the Fabric Links in the ACI fabric.
- Switched Port Analyzer (SPAN) sessions are configured, or TAPs placed in the VXLAN overlay network.

Supported PIDs to Strip VXLAN and iVXLAN

Beginning with Cisco NX-OS Release 10.2(2)F, the VXLAN stripping feature is supported on Cisco Nexus 9364C and 9300-EX, 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, 9300-GX2, 9500-EX, and 9500-FX line cards.

Beginning with Cisco NX-OS Release 10.2(2)F, the iVXLAN stripping feature is supported on Cisco Nexus 9364C and 9300-EX, 9300-FX, 9300-FX2, 9300-GX, 9300-GX2, 9500-EX, and 9500-FX line cards.

Guidelines and Limitations for VXLAN and iVXLAN Header Strip

- VXLAN header strip is supported when VXLAN underlay is V4.
- You must be able to strip VXLAN, and iVXLAN headers without being PTEP/VTEP.
- VXLAN header strip is enabled per port.
- VXLAN and iVXLAN strip is not supported if the following features are enabled:
 - NV overlay
 - VN-segment-vlan
 - Legacy MPLS strip and tap-aggregation

- VXLAN stripping is supported when the default UDP value is used.
- Ports must be able to manage both tunneled and non-tunneled packets.
- Layer 2 switch port mode trunk or Layer 2 PO interfaces must be able to strip the VXLAN header.
- Ensure that the Tap-ACL contains proper ACE with redirect keyword, where the redirect interfaces are pointing toward the egress/analyzer ports, else the packet will be flooded back on the same ingress port.
- OFM enables VXLAN strip capability for standard ISSU and LXC-ISSU.
- Beginning with Cisco NX-OS Release 10.2(1)F, the VXLAN and iVXLAN stripping features are supported on Cisco Nexus 9364C and 9300-EX, 9300-FX, 9300-FX2, 9500-EX, and 9500-FX line cards.
- Beginning with Cisco NX-OS Release 10.2(2)F, the VXLAN and iVXLAN stripping features are supported on Cisco Nexus 9300-GX and 9300-GX2 platform switches.
- A maximum of 4 tunnel-profiles can be created on a switch, one per type of encapsulation. However, beginning with Cisco NX-OS Release 10.2(3)F, a maximum of 5 tunnel-profiles are supported.
- A maximum of 12 redirect interfaces (prior to Release 10.2(1)) and 32 redirect interfaces (Release 10.2(1) and later) can only be configured in a single ACE of the TAP aggregation policy.
- For Cisco Nexus 9300-GX platform switches, post VXLAN strip, L2 header addresses are re-written as follows: Source MAC as VDC MAC address and Destination MAC as 000000abcdef.
- Beginning with Cisco NX-OS Release 10.2(3)F, VXLAN strip is supported on Cisco N9K-C93180YC-FX3 and N9K-C93108TC-FX3P platform switches.
- Beginning with Cisco NX-OS Release 10.2(4)M, the iVXLAN stripping feature is supported on Cisco N9K-C93180YC-FX3 and N9K-C93108TC-FX3P platform switches.

The below statements are true for post VXLAN, and iVXLAN header strip:

- The interface will allow slapping Q-in-Q VLAN on inside packet.
- Packet CRC will be properly performed.
- Inside packets will be allowed to filter using ingress port ACLs.

Configuring Nexus Data Broker Termination

The following steps outline the termination of NDB for VXLAN. The same procedure is followed for iVXLAN header strip.



Note

To change encapsulate tunnel type from VXLAN to iVXLAN or vice versa, the configured tunnel must be removed using no encapsulate CLI.



Note Ensure that the below CLIs are configured to enable stripping of VXLAN or iVXLAN on interfaces:

- destination any
- encapsulation vxlan
- flow terminate interface add Ethernet 1/1

If any of the above CLIs are missing, stripping of VXLAN or iVXLAN will not happen on the ports specified in flow term CLI.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal	Enters global configuration mode.
Step 2	feature ofm Example: switch (config)# feature ofm	Enables feature ofm.
Step 3	tunnel-profile profile-name Example: switch(config)# tunnel-profile vtep_vxlan_term switch(config-tnl-profile)#	Enables static VXLAN tunnels.
Step 4	encapsulation vxlan Example: switch(config-tnl-profile)# encapsulation vxlan switch(config-tnl-profile)#	To set appropriate encapsulation type for the tunnel profile.
Step 5	destination any Example: switch(config-tnl-profile)# destination any	To set required destination for the tunnel profile.
Step 6	flow terminate interface ethernet 1/1 Example: switch(config-tnl-profile)# flow terminate interface ethernet 1/1	To add ethernet1/1 to the flow term list (if the no flow terminate interface command was configured).
Step 7	flow terminate interface remove ethernet 1/1 Example:	To remove Ethernet 1/1 port only.

	Command or Action	Purpose
	<code>switch(config-tnl-profile)# flow terminate interface remove ethernet 1/1</code>	
Step 8	flow terminate interface add ethernet 1/2-5 Example: <code>switch(config-tnl-profile)# flow terminate interface add ethernet 1/2-5</code>	To add e1/2, e1/3, e1/4, e1/5 to an existing list of flow terminate interfaces. Note While adding flow terminate interface, CLI doesn't check whether L2 port interface exists or enabled. For example, e1/10 is a non-breakout mode. CLI allows interface e1/10/1-4 to add for flow terminate list. When e1/10 is a breakout, VXLAN header strip feature functions.
Step 9	flow terminate interface add port-channel 100-110 Example: <code>switch(config-tnl-profile)# flow terminate interface add po100-110</code>	To add port channel 100-110 to old list. New list will be e1/10-11 and po100-110.
Step 10	no flow terminate interface Example: <code>switch(config-tnl-profile)# no flow terminate interface</code>	To remove all flow and terminate interfaces from profile.
Step 11	feature tap-aggregation Example: <code>switch(config)# feature tap-aggregation</code>	Enables feature tap-aggregation.
Step 12	ip access-list <access-list name> Example: <code>switch(config)# ip access-list test switch(config-acl)#</code>	Creates an IPACL and enters the IP access list configuration mode.
Step 13	[no] permit protocol source destination redirect interfaces Example: <code>permit ip any any redirect interface ethernet 1/1, ethernet 1/19</code>	Creates an IP ACL rule that permits traffic to be redirected per its conditions. The no version of this command removes the permit rule from the policy. Note When you enter an interface for the TAP aggregation policy, do not abbreviate it. When you enter a list of interfaces, separate them with commas, but no spaces.
Step 14	ip port access-group <access-group name> in Example:	Applies the port access list to the ERSPAN strip/terminating port.

	Command or Action	Purpose
	configure terminal	
	interface Ethernet 1/32	
	ip port access-group test in	

Configuration Example for VXLAN and iVXLAN Header Strip

The following example shows VXLAN and iVXLAN header stripping, the procedure is same for iVXLAN:

```
switch(config-tnl-profile)# show run ofm
show running-config ofm
feature ofm
tunnel-profile vxlan1
encapsulation vxlan
destination any
flow terminate interface add port-channel101
flow terminate interface add Ethernet1/1

tunnel-profile vxlan2
encapsulation ivxlan
destination any
flow terminate interface add port-channel101
flow terminate interface add Ethernet1/1
switch(config-tnl-profile)#
switch(config-tnl-profile)# show tunnel-profile
Profile : vxlan1
Encapsulation : Vxlan
State : UP
Destination : Any
Terminate Interfaces : 2
Terminate List : port-channel101 Ethernet1/1
Profile : vxlan2
Encapsulation : iVxlan
State : UP
Destination : Any
Terminate Interfaces : 2
Terminate List : port-channel101 Ethernet1/1
switch(config-tnl-profile)#
```

ERSPAN Header Stripping for Nexus Data Broker

This subchapter describes ERSPAN header stripping procedure for Cisco Nexus platform switch. The primary use case for this is on Nexus Data Broker (NDB) switch.

This chapter contains the following sections:

About ERSPAN Header Stripping

This feature implements inline ERSPAN header stripping from the incoming ERSPAN packets on NX-OS switch or Nexus Data Broker (NDB) switch.

When the ERSPAN packets come in, this feature strips the ERSPAN header and forwards it to the outside box inline, that is, a packet comes on to a terminating port, and then, based on the ACL configuration, it is redirected to the ports that are connected to the outside server.

This feature does a single pass ERSPAN header stripping and PACL redirect.

Supported PIDs to Strip the ERSPAN Header

Beginning with Cisco NX-OS Release 10.2(1)F, ERSPAN header stripping is supported on Cisco Nexus 9300-FX2, 9300-FX3, 9300-GX, and 9300-GX2 platform switches. However, this feature is only supported on TOR switches.

Guidelines and Limitations for ERSPAN Header Stripping

- The incoming port must be a layer 2 port, but its connectivity to layer 3 must be through SVI.
- ERSPAN destination session and ERSPAN stripping cannot co-exist.
- The total number of terminating ports including port channel members cannot be more than 31.
- Mode tap-aggr should not be configured for this feature.
- Tunnel profile for all ERSPAN ID is supported. Termination of specific ERSPAN session ID is not supported. Traffic with any ERSPAN session ID will be terminated at the termination node.
- Only 1 tunnel profile per node is supported.
- A maximum of 31 flow terminate interfaces are supported on tunnel-profile with encap type: ERSPAN.
- The ERSPAN header stripping feature is supported on Cisco Nexus 9300-FX2, 9300-FX3, 9300-GX, and 9300-GX2 platform switches. Note that this feature is only supported on TOR switches.
- You need to enable ERSPAN stripping on the port so that ERSPAN strip/redirect works properly. Do not send ERSPAN traffic on ports where other strips are enabled.
- Strips all the incoming ERSPAN headers on the terminating port.
- This feature works only when OFM tunnel profiles and ACL redirect are configured.
- This feature will work only when port ACL is applied to the layer 2 terminating port.
- There can be only one tunnel profile for ERSPAN encapsulation on the switch.
- Appropriate tcam needs to be carved to use port acl, for example, **tcam region ing-ifacl** should be used for carving.

Configuring ERSPAN Header Stripping

The following steps outline the configuration for ERSPAN header stripping.



Note Ensure that the below CLIs are configured to enable stripping of ERSPAN on interfaces:

- encapsulation erspan
- erspan session-id all
- flow terminate interface add e1/16

If any of the above CLIs are missing, stripping of ERSPAN does not happen on the ports specified in flow term CLI.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal	Enters global configuration mode.
Step 2	feature ofm Example: switch (config)# feature ofm	Enables feature ofm.
Step 3	tunnel-profile <profile-name> Example: switch(config)# tunnel-profile foo switch(config-tnl-profile)#	Enables static ERSPAN tunnels.
Step 4	encapsulation erspan Example: switch(config-tnl-profile)# encapsulation erspan switch(config-tnl-profile)#	To set appropriate encapsulation type for the tunnel profile.
Step 5	erspan session-id all Example: switch(config-tnl-profile)# erspan session-id all	The ERSPAN session ID denotes the monitored session that the related ERSPAN packet is associated with on the source switch.
Step 6	flow terminate interface add ethernet1/16 Example: switch(config-tnl-profile)# flow terminate interface add ethernet1/16	To add ethernet1/16 to the flow term list (if no flow CLI is configured).
Step 7	ip access-list <access-list-name> Example:	Creates an IPACL and enters the IP access list configuration mode.

	Command or Action	Purpose
	switch(config)# ip access-list test switch(config-acl)#	
Step 8	[no] permit protocol source destination redirect interfaces Example: <pre>permit ip any any redirect ethernet1/1,ethernet1/19</pre>	Creates an IP ACL rule that permits traffic to be redirected per its conditions. The no version of this command removes the permit rule from the policy. Note When you enter an interface for the TAP aggregation policy, do not abbreviate it. When you enter a list of interfaces, separate them with commas, but no spaces.
Step 9	ip port access-group <access-group name> _redir in Example: <pre>interface e1/16 (config-if)# ip port access-group test in</pre>	Applies the port access list to the ERSPAN strip/terminating port.

Configuration Example for ERSPAN Header Stripping

The following example shows ERSPAN header stripping:

```
switch(config)# feature ofm
switch(config)# tunnel-profile foo
switch(config-tnl-profile)# encapsulation erspan
switch(config-tnl-profile)# erspan session-id all
switch(config-tnl-profile)# flowterminate interface add ethernet1/16
switch(config)# ip access-list test
permit ip any any redirect ethernet1/1,ethernet1/19
interface e1/16 (config-if)# ip port access-group test in
```

Verifying the Configuration for ERSPAN Header Stripping

To display the ERSPAN header stripping configuration, perform one of the following tasks:

Command	Purpose
show run ofm	Displays the tunnel profiles.
show run aclmgr	Displays all the ACLs and the application of those ACLs on the interfaces.
show ip access-list acl_nam	Displays ACL hit and redirected packets count.
show tunnel-profile	Displays the states of all tunnel profiles.

GRE Header Stripping for Nexus Data Broker

This subchapter describes GRE header stripping procedure for Cisco Nexus platform switch. The primary use case for this is on Nexus Data Broker (NDB) switch.

This chapter contains the following sections:

About NDB GRE Header Stripping

This feature allows you to strip the GRE header from packets that come in with a GRE encapsulation. The inner packet in a GRE encapsulated packet does not contain an ethernet header. So, after a GRE strip, an ethernet header is added to the inner packet with the following custom fields:

1. 802.1q header with vlan configured on the incoming port.
2. Destination MAC address will be set to 00:00:00:ab:cd:ef or 000.000.abc.def.
3. Source MAC address will be set to VDC MAC address of the switch.

NDB GRE Header Stripping Guidelines and Limitations

- To remove flow interface from a tunnel-profile, use **remove** instead of **no**. The use of **no** in flow terminate command will delete all interfaces from flow terminate list.

For example:

```
switch(config)# tunnel-profile gre_strip  
switch(config-tnl-profile)# flow terminate interface remove Ethernet 1/48
```

- Flow terminate interfaces cannot share ESPRAN and GRE/VXLAN/IVXLAN profiles.
- If GRE strip-enabled interface receives ERSPAN traffic, stripping succeeds, but traffic will not be forwarded to the redirect port.
- Feature OFM and feature tunnel cannot co-exist on the same switch.
- The NDB GRE Header Stripping feature is supported on Cisco Nexus 9300-EX, 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, and N9K-C9332D-GX2B TORs. However, this feature is not supported on line cards.
- The configuration of **mode tap-aggregation** should not be present on interface where GRE header stripping functionality is enabled.
- Tunnel-encapsulation type modification is not allowed.

```
QP-CF-1(config-tnl-profile)# encapsulation gre  
Error: encap-type modify not allowed, delete and add again
```
- A maximum of 500 flow terminate interfaces are supported on tunnel-profile with encap type: iVXLAN/VXLAN/GRE.
- A maximum of 31 flow terminate interfaces are supported on tunnel-profile with encap type: ERSPAN.

- When flow terminate interface CLI is configured without **add** keyword, it acts as **replace**, which means previously added flow terminate interfaces are deleted and only new ones will act as flow terminate interfaces.
- After non-disruptive upgrade from previous NX-OS version to 10.2(3)F, port ACL must be removed from all interfaces and added before enabling GRE header strip feature for particular interface.
- The **hardware acl tap-aggregate redirect disable-dot1q-sharing** command is required on 9300-GX to allow dot1q tunnel propagation. The switch needs reload after enabling this command.

CLIs for GRE Header Strip Feature

The following are the CLIs to be configured for enabling GRE header on an interface:

```
feature ofm
tunnel-profile gre_strip
    encapsulation gre
    destination any
    flow terminate interface add Ethernet1/1-10
```

The following is the show command for tunnel-profile:

```
switch# show tunnel-profile gre_strip
Profile           : gre_strip
Encapsulation     : GRE
State             : UP
Destination       : Any
Terminate Interfaces : 10
Terminate List    : Ethernet1/1 Ethernet1/2 Ethernet1/3 Ethernet1/4 Ethernet1/5
Ethernet1/6 Ethernet1/7 Ethernet1/8 Ethernet1/9 Ethernet1/10
```

Configuration for Egress and Ingress Ports

The following is the configuration for ingress ports:

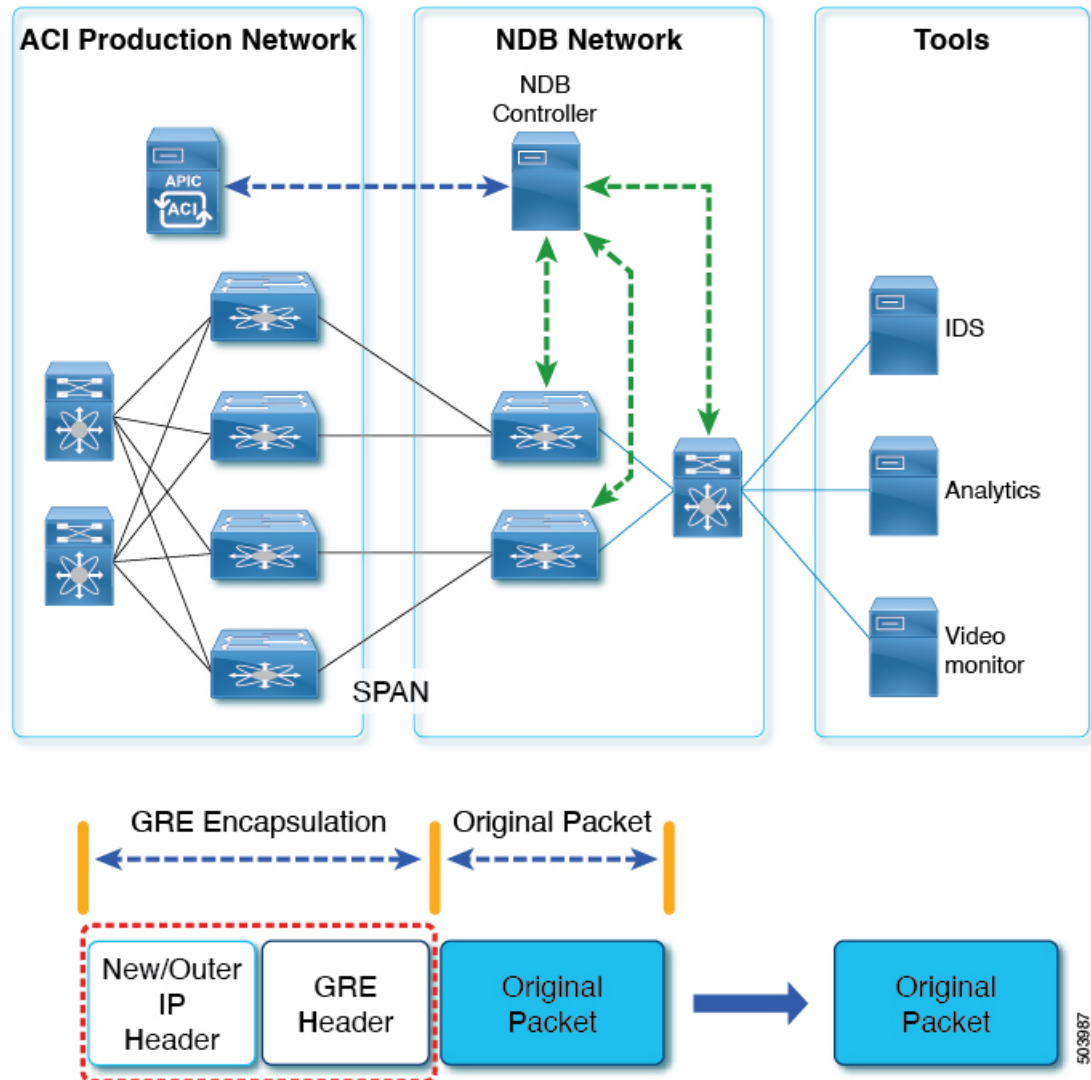
```
interface eth1/1
    switchport access vlan 101
    switchport mode dot1q-tunnel
    ip port access-group ndb_acl in <<<
    no shutdown
```

The following is the configuration for egress ports:

```
interface Ethernet1/7
    switchport mode trunk
    no shutdown

IP access list ndb_acl
    statistics per-entry
    10 permit udp any any eq 4789 redirect Ethernet1/7
    15 permit ip any any redirect Ethernet1/7
```

Figure 9: NDB GRE Header Strip Solution



MPLS Header Stripping for Nexus Data Broker

This subchapter describes MPLS header stripping procedure for Cisco Nexus platform switch. The primary use case for this is on Nexus Data Broker (NDB) switch.

This chapter contains the following sections:

About NDB MPLS Header Stripping

This feature allows you to strip the MPLS header from packets that come in with a MPLS encapsulation. MPLS label stripping is supported for both IPoMPLS and EoMPLS packet formats. After MPLS label strip, an ethernet header is added to the inner packet with the following custom fields:

1. 802.1q header with vlan configured on the incoming port.
2. Destination MAC address will be set to 00:00:00:ab:cd:ef or 000.000.abc.def.



Note For EoMPLS header stripping, this is applicable only on Cisco Nexus 9300-EX, 9300-FX, and 9300-GX platforms.

3. Source MAC address will be set to VDC MAC address of the switch.



Note For EoMPLS header stripping, this is applicable only on Cisco Nexus 9300-EX, 9300-FX, and 9300-GX platforms.

NDB MPLS Header Stripping Guidelines and Limitations

The following guidelines and limitations apply when migrating from legacy MPLS header stripping to OFM-based configuration:

- Legacy MPLS stripping implementation cannot co-exist with any OFM-based stripping.
- Feature OFM and feature tunnel cannot co-exist on the same switch.
- Migrating from legacy MPLS stripping functionality requires the following cleanup before enabling OFM-based MPLS stripping:
 - Removal of **mode tap-aggregation** at interface(s) level
 - Removal of **mpls strip; mpls strip dot1q** at the global level
 - Save the configuration and reload the switch with the above configuration
- Beginning with Cisco NX-OS Release 10.2(3)F, the NDB MPLS Header Stripping feature is supported.
 - IPoMPLS (packet format) header stripping is supported on Cisco Nexus 9300-EX, 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, and C9332D-GX2B platforms.
 - EoMPLS (packet format) header stripping is supported only on Cisco Nexus 9300-EX platform switches. However, VPLS strip and control-word packet strip is not supported.



Note The OFM MPLS stripping feature is supported only on TORs; it is not supported on line cards.

- After non-disruptive upgrade from previous NX-OS version to 10.2(3)F, port ACL must be removed from all interfaces and added before enabling MPLS header stripping feature for a particular interface.
- The **hardware acl tap-agg redirect disable-dot1q-sharing** command is required on Cisco Nexus 9300-GX platform switches to allow dot1q tunnel propagation. The switch needs reload after enabling this command.

- Tunnel-encapsulation type modification is not allowed.

```
QP-CF-1(config-tnl-profile)# encapsulation mpls
Error: encap-type modify not allowed, delete and add again
```

- If ERSPAN ACL redirect tunnel-profile is not configured and the interface is receiving ERSPAN packets, then the ERSPAN packets will hit ERSPAN ACL redirect entries in TapAgg policy and will not be stripped.
- On an interface where MPLS head strip is enabled, mode tap-aggregation should not be present.
- MPLS Stripping is based on IP PACL, so do not use MAC-ACL for stripping.
- During MPLS stripping, incoming VLAN in the original packet is not preserved.
- With ERSPAN tunnel-profile, when ingress interface is converted from dot1q-tunnel to trunk mode, egress packets will have dot1q tag with VLAN=1. This tagging takes place for both stripped packets and regular IP packets that are redirected.
- When an MPLS strip-enabled interface receives ERSPAN traffic, stripping succeeds, but traffic is not forwarded to the redirect port.
- To remove flow interface from a tunnel-profile, use **remove** instead of **no**. The use of **no** in flow terminate command will delete all interfaces from flow terminate list.

For example:

```
switch(config)# tunnel-profile mpls_strip
switch(config-tnl-profile)# flow terminate interface remove Ethernet 1/48
```

- When flow terminate interface command is configured without the **add** keyword, it acts as **replace**, which means previously added flow terminate interfaces are deleted and only the new ones will act as flow terminate interfaces.
- MPLS packets with multicast bit set cannot be terminated, but MPLS packets with unicast bit set can be terminated.
- Ingress interface can be either in trunk mode or access mode. Both modes allow redirection of tagged and untagged packets. When access-mode is used along with dot1q-tunnel mode, after header stripping VLAN_tag is added as specified by the access-mode.
- Until Cisco NX-OS Release 10.3(1)F, EoMPLS header stripping was supported only on Cisco Nexus 9300-EX platform switches (VPLS strip and control-word packet strip were not supported). Beginning with Cisco NX-OS Release 10.3(2)F, EoMPLS header stripping feature is also supported on Cisco Nexus 9300-FX, 9300-FX2, 9300-FX3, 9300-GX, and 9300-GX2 ToRs; it is not supported on line cards. The following guidelines and limitations are applicable:
 - EoMPLS stripping can co-exist with all other header stripping features on same or different interfaces.
 - For Cisco Nexus 9300-EX, 9300-FX, and 9300-GX platform switches, post EoMPLS header strip, L2 header addresses are re-written as follows: Source MAC as VDC MAC address and Destination MAC as 000000abcdef.
 - Pseudo Wire Control Word is not supported.
 - On Cisco Nexus 9300-GX platform switches, two ingress ports cannot share acl unless the dot1q vlan config is the same on them, else tagging does not work.

Commands for MPLS Header Strip Feature

The following commands should be configured for enabling MPLS header on an interface:

```
feature ofm
tunnel-profile
mpls_strip encapsulation mpls destination any
flow terminate interface add Ethernet1/1-10
```

The show command for tunnel-profile is as follows:

```
switch# show tunnel-profile mpls_strip
Profile           : mpls_strip
Encapsulation     : MPLS
State             : UP
Destination       : Any
Terminate Interfaces : 10
Terminate List     : Ethernet1/1 Ethernet1/2 Ethernet1/3 Ethernet1/4 Ethernet1/5
Ethernet1/6 Ethernet1/7 Ethernet1/8 Ethernet1/9 Ethernet1/10
```

Configuration for Egress and Ingress Ports

The following is the configuration for ingress ports:

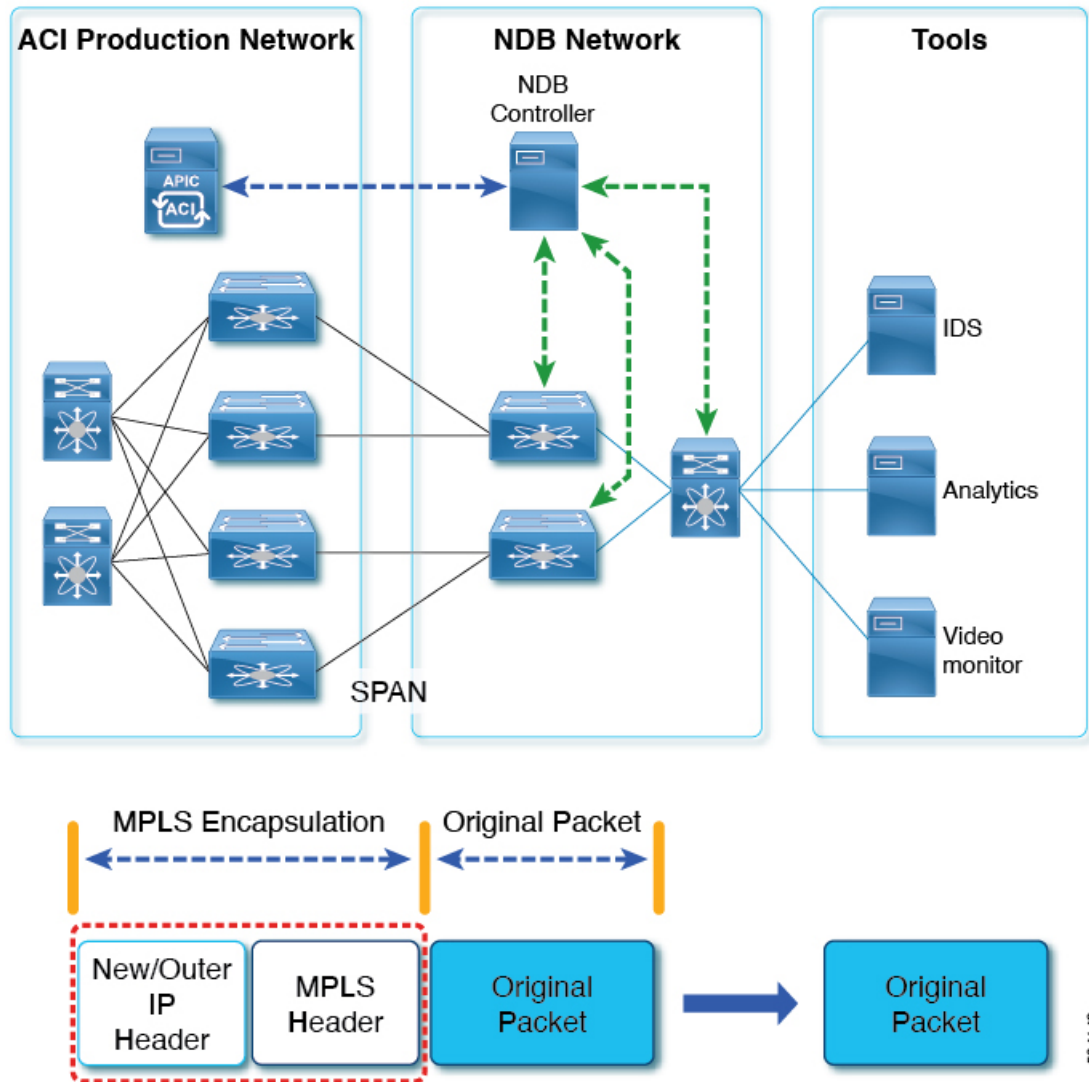
```
interface eth1/1
  switchport access vlan 101
  switchport mode dot1q-tunnel
  ip port access-group ndb_acl in
  no shutdown
```

The following is the configuration for egress ports:

```
interface Ethernet1/7
  switchport mode trunk
  no shutdown

IP access list ndb_acl
  statistics per-entry
  10 permit udp any any eq 4789 redirect Ethernet1/7
  15 permit ip any any redirect Ethernet1/7
```

Figure 10: NDB MPLS Header Strip Solution

**Note**

In case of decapsulated packet such as MPLS, the NDB-switch adds an Ethernet/VLAN header to the **original packet**, so egressing packet will have Ethernet/VLAN - original packet.



CHAPTER 31

Configuring Graceful Insertion and Removal

This chapter describes how to configure graceful insertion and removal (GIR) on the Cisco Nexus 9000 Series switches.

This chapter contains the following sections:

- [About Graceful Insertion and Removal, on page 505](#)
- [Guidelines and Limitations for GIR, on page 507](#)
- [GIR Workflow, on page 508](#)
- [Configuring the Maintenance-Mode Profile, on page 509](#)
- [Configuring the Normal-Mode Profile, on page 510](#)
- [Creating a Snapshot, on page 512](#)
- [Adding Show Commands to Snapshots, on page 513](#)
- [Triggering Graceful Removal, on page 514](#)
- [Triggering Graceful Insertion, on page 517](#)
- [Maintenance Mode Enhancements, on page 519](#)
- [Verifying the GIR Configuration, on page 520](#)
- [Configuration Examples for GIR, on page 521](#)

About Graceful Insertion and Removal

You can use graceful insertion and removal to gracefully eject a switch and isolate it from the network in order to perform debugging or upgrade operations. The switch is removed from the regular forwarding path with minimal traffic disruption. When you are finished performing debugging or upgrade operations, you can use graceful insertion to return the switch to its fully operational (normal) mode.

When you place the switch in maintenance mode, all configured Layer 3 control-plane protocols are isolated from the network. Directly connected routes are not withdrawn or modified during this state. When normal mode is restored, the advertisement of all routes is restored.

In graceful removal, all protocols and vPC domains are gracefully brought down and the switch is isolated from the network. In graceful insertion, all protocols and vPC domains are restored.

The following protocols are supported (for both IPv4 and IPv6 address families):

- Border Gateway Protocol (BGP)
- Enhanced Interior Gateway Routing Protocol (EIGRP)
- Intermediate System-to-Intermediate System (ISIS)

- Open Shortest Path First (OSPF)
- Protocol Independent Multicast (PIM)
- Routing Information Protocol (RIP)



Note For graceful insertion and removal, the PIM protocol is applicable only to vPC environments. During graceful removal, the vPC forwarding role is transferred to the vPC peer for all northbound sources of multicast traffic.

Profiles

By default, the system isolates all enabled protocols during graceful removal and restores them during graceful insertion. The protocols are isolated and restored in a predefined order.

If you want to isolate, shut down, or restore the protocols individually (or perform additional configurations), you can create a profile with configuration commands that can be applied during graceful removal or graceful insertion. However, you need to make sure that the order of the protocols is correct and any dependencies are considered.

The switch supports the following profiles:

- Maintenance-mode profile—Contains all the commands that will be executed during graceful removal, when the switch enters maintenance mode.
- Normal-mode profile—Contains all the commands that will be executed during graceful insertion, when the switch returns to normal mode.

The following commands (along with any configuration commands) are supported in the profiles.



Note The **shutdown** command takes precedence when both **shutdown** and **isolate** are configured under a routing protocol instance or maintenance-mode profile.

Command	Description
isolate	Isolates the protocol from the switch and puts the protocol in maintenance mode.
no isolate	Restores the protocol and puts the protocol in normal mode.
shutdown	Shuts down the protocol or vPC domain.
no shutdown	Brings up the protocol or vPC domain.
system interface shutdown [exclude fex-fabric]	Shuts down the system interfaces (except the management interface).
no system interface shutdown [exclude fex-fabric]	Brings up the system interfaces.

Command	Description
sleep instance <i>instance-number seconds</i>	Delays the execution of the command by a specified number of seconds. You can delay multiple instances of the command. The range for the <i>instance-number</i> and <i>seconds</i> arguments is from 0 to 2177483647.
python instance <i>instance-number uri [python-arguments]</i> Example: python instance 1 bootflash://script1.py	Configures Python script invocations to the profile. You can add multiple invocations of the command to the profile. You can enter a maximum of 32 alphanumeric characters for the Python arguments.



Note Beginning with Cisco NX-OS Release 9.3(5), the **isolate** command is provided with the **include-local** option, which is applicable only to **router bgp**.

If you use this option, BGP withdraws all the routes from its peers. If you do not use this option, then BGP only withdraws remotely learned routes, and the locally originated routes such as aggregate, injected, network and redistribute continue to be advertised with maximum Multi-Exit Discriminator (MED) to eBGP peers and minimum local preference to iBGP peers.

Snapshots

In Cisco NX-OS, a snapshot is the process of capturing the running states of selected features and storing them on persistent storage media.

Snapshots are useful to compare the state of a switch before graceful removal and after graceful insertion. The snapshot process consists of three parts:

- Creating a snapshot of the states of a few preselected features on the switch and storing them on the persistent storage media
- Listing the snapshots taken at various time intervals and managing them
- Comparing snapshots and showing the differences between features

Guidelines and Limitations for GIR

Graceful Insertion and Replacement have the following configuration guidelines and limitations:

- Beginning with Cisco NX-OS Release 9.2(1), L2 Graceful Insertion and Replacement is supported. When moving from normal to maintenance mode, MCT goes down resulting in north to south traffic convergence.

Zero packet loss is not supported. The following table provides an example of traffic convergence of 10 vPCs with 2 port member on each VPC port and 60k mac scale.

Table 26:

Trigger	Role	North to South Traffic	South to North Traffic
Normal to maintenance mode	Primary	760 ms	1320 ms
Maintenance mode to normal	Primary	13155 ms	27980 ms
Normal to maintenance mode	Secondary	300 ms	1375 ms
Maintenance mode to normal	Secondary	15905 ms	23350 ms

- Beginning with Cisco NX-OS Release 9.2(1), if you configure the **isolate** option for OSPF, direct routes and stub routes are advertised as max-metric routes. As a result, north-to-south traffic to the SVI hosts goes through the vPC peer when only one vPC switch is isolated.
- Remove all existing custom profiles before creating new custom profiles for normal-mode and maintenance-mode.
- Beginning with Cisco NX-OS Release 9.3(5), the **include-local** option is added to the existing **isolate** command. However, the **include-local** option applies only to **router bgp**.
- Beginning with Cisco NX-OS Release 10.3(1)F, during a binary reload scenario, when the system is switched to maintenance mode, the interface is shut until the system boots up completely and declares system readiness.

GIR Workflow

Follow these steps to complete the graceful insertion and removal (GIR) workflow:

1. (Optional) Create the maintenance-mode profile. (See [Configuring the Maintenance-Mode Profile, on page 509](#).)
2. (Optional) Create the normal-mode profile. (See [Configuring the Normal-Mode Profile, on page 510](#).)
3. Take a snapshot before triggering graceful removal. (See [Creating a Snapshot, on page 512](#).)
4. Trigger graceful removal to put the switch in maintenance mode. (See [Triggering Graceful Removal, on page 514](#).)
5. Trigger graceful insertion to return the switch to normal mode. (See [Triggering Graceful Insertion, on page 517](#).)
6. Take a snapshot after triggering graceful insertion. (See [Creating a Snapshot, on page 512](#).)

7. Use the **show snapshots compare** command to compare the operational data before and after the graceful removal and insertion of the switch to make sure that everything is running as expected. (See [Verifying the GIR Configuration, on page 520.](#))

Configuring the Maintenance-Mode Profile

You can create a maintenance-mode profile with configuration commands that can be applied during graceful removal or graceful insertion.



Note During the maintenance mode the SVI becomes UP after the reload. In this scenario, use the **isolate include-local** command under router BGP or keep interfaces in shutdown state through maintenance mode to avoid the impact of advertising connected/static routes.

Procedure

	Command or Action	Purpose
Step 1	[no] configure maintenance profile maintenance-mode Example: <pre>switch# configure maintenance profile maintenance-mode Enter configuration commands, one per line. End with CNTL/Z. switch(config-mm-profile) #</pre>	Enters a configuration session for the maintenance-mode profile. The no option deletes the maintenance profile maintenance-mode. Depending on which protocols you have configured, you must now enter the appropriate commands to bring down the protocols. For a list of supported commands, see Profiles, on page 506.
Step 2	end Example: <pre>switch(config-mm-profile) # end switch#</pre>	Closes the maintenance-mode profile.
Step 3	show maintenance profile maintenance-mode Example: <pre>switch# show maintenance profile maintenance-mode</pre>	Displays the details of the maintenance-mode profile.

Example

This example shows how to create a maintenance-mode profile:

```
switch# configure maintenance profile maintenance-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile) # ip pim isolate
switch(config-mm-profile) # router bgp 100
```

```

switch(config-mm-profile-router)# shutdown
switch(config-mm-profile)# router eigrp 10
switch(config-mm-profile-router)# shutdown
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# shutdown
switch(config-mm-profile)# vpc domain 10
switch(config-mm-profile-config-vpc-domain)# shutdown
switch(config-mm-profile)# system interface shutdown
switch(config-mm-profile)# end
Exit maintenance profile mode.
switch# show maintenance profile maintenance-mode
[Maintenance Mode]
ip pim isolate
router bgp 100
    shutdown
router eigrp 10
    shutdown
    address-family ipv6 unicast
        shutdown
vpc domain 10
    shutdown
system interface shutdown

```

This example shows how to configure sleep instance in a custom profile to add a delay before the next protocol change.

```

switch# configure maintenance profile maintenance-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# router bgp 65001
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# sleep instance 1 10
switch(config-mm-profile)# router eigrp 200
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# sleep instance 2 15
switch(config-mm-profile)# router ospf 100
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# sleep instance 3 20
switch(config-mm-profile)# router ospfv3 300
switch(config-mm-profile-router)# isolate
switch(config-mm-profile-router)# sleep instance 4 5
switch(config-mm-profile)# router isis 400
switch(config-mm-profile-router)# isolate
switch(config-mm-profile)#end
Exit maintenance profile mode.
switch#

```



Note If you need to run exec commands or add a dynamic delay while the maintenance mode profile is applied, use the **python instance** *instance-number uri [python-arguments]* script.

Configuring the Normal-Mode Profile

You can create a normal-mode profile with configuration commands that can be applied during graceful removal or graceful insertion.

Procedure

	Command or Action	Purpose
Step 1	[no] configure maintenance profile normal-mode Example: <pre>switch# configure maintenance profile normal-mode Enter configuration commands, one per line. End with CNTL/Z. switch(config-mm-profile) #</pre>	Enters a configuration session for the normal-mode profile. The no version removes the maintenance profile normal-mode. Depending on which protocols you have configured, you must now enter the appropriate commands to bring up the protocols. For a list of supported commands, see Profiles, on page 506.
Step 2	end Example: <pre>switch(config-mm-profile) # end switch#</pre>	Closes the normal-mode profile.
Step 3	show maintenance profile normal-mode Example: <pre>switch# show maintenance profile normal-mode</pre>	Displays the details of the normal-mode profile.

Example

This example shows how to create a maintenance profile normal-mode:

```
switch# configure maintenance profile normal-mode
switch(config-mm-profile) # no system interface shutdown
switch(config-mm-profile) # vpc domain 10
switch(config-mm-profile-config-vpc-domain) # no shutdown
switch(config-mm-profile) # router eigrp 10
switch(config-mm-profile-router) # no shutdown
switch(config-mm-profile-router) # address-family ipv6 unicast
switch(config-mm-profile-router-af) # no shutdown
switch(config-mm-profile) # router bgp 100
switch(config-mm-profile-router) # no shutdown
switch(config-mm-profile) # no ip pim isolate
switch(config-mm-profile) # end
Exit maintenance profile mode.
switch# show maintenance profile normal-mode
[Normal Mode]
no system interface shutdown
vpc domain 10
  no shutdown
  router eigrp 10
    no shutdown
address-family ipv6 unicast
  no shutdown
router bgp 100
  no shutdown
no ip pim isolate
```

Creating a Snapshot

You can create a snapshot of the running states of selected features. When you create a snapshot, a predefined set of **show** commands are run and the outputs are saved.

Procedure

	Command or Action	Purpose
Step 1	snapshot create <i>snapshot-name description</i> Example: <pre>switch# snapshot create snap_before_maintenance Taken before maintenance Executing 'show interface'... Done Executing 'show ip route summary vrf all'... Done Executing 'show ipv6 route summary vrf all'... Done Executing 'show bgp sessions vrf all'... Done Executing 'show ip eigrp topology summary'... Done Executing 'show ipv6 eigrp topology summary'... Done Feature 'vpc' not enabled, skipping... Executing 'show ip ospf vrf all'... Done Feature 'ospfv3' not enabled, skipping... Feature 'isis' not enabled, skipping... Feature 'rip' not enabled, skipping... Snapshot 'snap_before_maintenance' created</pre>	Captures the running state or operational data of selected features and stores the data on persistent storage media. You can enter a maximum of 64 alphanumeric characters for the snapshot name and a maximum of 254 alphanumeric characters for the description. Use the snapshot delete {all <i>snapshot-name</i>} command to delete all snapshots or a specific snapshot.
Step 2	show snapshots Example: <pre>switch# show snapshots Snapshot Name Time Description ----- snap_before_maintenance Wed Aug 19 13:53:28 2015 Taken before maintenance</pre>	Displays snapshots present on the switch.
Step 3	show snapshots compare <i>snapshot-name-1 snapshot-name-2</i> [summary ipv4routes ipv6routes] Example: <pre>switch# show snapshots compare snap_before_maintenance snap_after_maintenance</pre>	Displays a comparison of two snapshots. The summary option displays just enough information to see the overall changes between the two snapshots. The ipv4routes and ipv6routes options display the changes in IPv4 and IPv6 routes between the two snapshots.

Example

The following example shows a summary of the changes between two snapshots:

```
switch# show snapshots compare snapshot1 snapshot2 summary
feature                               snapshot1  snapshot2  changed
basic summary
  # of interfaces                     16         12         *
  # of vlans                          10         4         *
  # of ipv4 routes                    33         3         *
.....

interfaces
  # of eth interfaces                 3          0         *
  # of eth interfaces up              2          0         *
  # of eth interfaces down            1          0         *
  # of eth interfaces other           0          0
  # of vlan interfaces                3          1         *
  # of vlan interfaces up             3          1         *
  # of vlan interfaces down           0          0
  # of vlan interfaces other          0          1         *
.....
```

The following example shows the changes in IPv4 routes between two snapshots:

```
switch# show snapshots compare snapshot1 snapshot2 ipv4routes
metric                               snapshot1  snapshot2  changed
# of routes                          33         3         *
# of adjacencies                     10         4         *

Prefix                               Changed Attribute
-----
23.0.0.0/8                           not in snapshot2
10.10.10.1/32                         not in snapshot2
21.1.2.3/8                           adjacency index has changed from 29 (snapshot1) to 38 (snapshot2)
.....
```

There were 28 attribute changes detected

Adding Show Commands to Snapshots

You can specify additional **show** commands to be captured in snapshots. These **show** commands are defined in user-specified snapshot sections.

Procedure

	Command or Action	Purpose
Step 1	snapshot section add section "show-command" row-id element-key1 [element-key2] Example: switch# snapshot section add myshow "show ip interface brief" ROW_intf intf-name	Adds a user-specified section to snapshots. The <i>section</i> is used to name the show command output. You can use any word to name the section.

	Command or Action	Purpose
		The show command must be enclosed in quotation marks. Non-show commands will not be accepted. The <i>row-id</i> argument specifies the tag of each row entry of the show command's XML output. The <i>element-key1</i> and <i>element-key2</i> arguments specify the tags used to distinguish among row entries. In most cases, only the <i>element-key1</i> argument needs to be specified to be able to distinguish among row entries. Note To delete a user-specified section from snapshots, use the snapshot section delete section command.
Step 2	show snapshots sections Example: switch# show snapshots sections	Displays the user-specified snapshot sections.

Example

The following example adds the **show ip interface brief** command to the myshow snapshot section.

```
switch# snapshot section add myshow "show ip interface brief" ROW_intf intf-name
switch# show snapshots sections
user-specified snapshot sections
-----
[myshow]
  cmd: show ip interface brief
  row: ROW_intf
  key1: intf-name
  key2: -

[sect2]
  cmd: show ip ospf vrf all
  row: ROW_ctx
  key1: instance_number
  key2: cname
```

Triggering Graceful Removal

In order to perform debugging or upgrade operations, you can trigger a graceful removal of the switch, which will eject the switch and isolate it from the network.

Before you begin

If you want the system to use a maintenance-mode profile that you create, see [Configuring the Maintenance-Mode Profile, on page 509](#).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	system mode maintenance [dont-generate-profile timeout value shutdown on-reload reset-reason reason] Example: <pre>switch(config)# system mode maintenance Following configuration will be applied: ip pim isolate router bgp 65502 isolate router ospf p1 isolate router ospfv3 p1 isolate Do you want to continue (y/n)? [no] y Generating a snapshot before going into maintenance mode Starting to apply commands... Applying : ip pim isolate Applying : router bgp 65502 Applying : isolate Applying : router ospf p1 Applying : isolate Applying : router ospfv3 p1 Applying : isolate Maintenance mode operation successful.</pre>	Puts all enabled protocols in maintenance mode (using the isolate command). The following options are available: • dont-generate-profile—Prevents the dynamic searching of enabled protocols and executes commands configured in a maintenance-mode profile. Use this option if you want the system to use a maintenance-mode profile that you have created. • timeout value—Keeps the switch in maintenance mode for a specified number of minutes. The range is from 5 to 65535. Once the configured time elapses, the switch returns to normal mode automatically. The no system mode maintenance timeout command disables the timer. • shutdown—Shuts down all protocols, vPC domains, and interfaces except the management interface (using the shutdown command). This option is disruptive while the default (which uses the isolate command) is not. • on-reload reset-reason reason—Boots the switch into maintenance mode automatically in the event of a specified system crash. The no system mode maintenance on-reload reset-reason command prevents the switch from being brought up in maintenance mode in the event of a system crash. The maintenance mode reset reasons are as follows: • HW_ERROR—Hardware error • SVC_FAILURE—Critical service failure

	Command or Action	Purpose
		• KERN_FAILURE—Kernel panic • WDOG_TIMEOUT—Watchdog timeout • FATAL_ERROR—Fatal error • LC_FAILURE—Line card failure • MATCH_ANY—Any of the above reasons Note • During reload, a binary form of configuration file is used to restore the system. However, this is not possible in all scenarios, for example, when the reloaded image is new. Then, the system switches to ASCII reload. • During an unexpected reload, when the system switches to maintenance mode, the interface will be in a shut state (reason: mmodeBootIntfShut), until the system comes up completely from normal mode to maintenance mode. • During reload, if the system does not come up completely for a duration that is longer than expected, then collect the debugging information using the following commands and contact Cisco Support: • show tech support mmode • show tech support system manager • show tech support interface manager • show accounting The system prompts you to continue. Enter y to continue or n to terminate the process.
Step 3	(Optional) show system mode Example: <pre>switch(config)# show system mode System Mode: Maintenance</pre>	Displays the current system mode. The switch is in maintenance mode. You can now perform any desired debugging or upgrade operations on the switch.

	Command or Action	Purpose
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration. This command is required if you want to preserve maintenance mode following a reboot.

Example

This example shows how to shut down all protocols, vPC domains, and interfaces on the switch:

```
switch(config)# system mode maintenance shutdown
```

Following configuration will be applied:

```
vpc domain 10
  shutdown
router bgp 65502
  shutdown
router ospf p1
  shutdown
router ospfv3 p1
  shutdown
system interface shutdown
```

Do you want to continue (y/n)? [no] **y**

Generating a snapshot before going into maintenance mode

Starting to apply commands...

```
Applying : vpc domain 10
Applying :   shutdown
Applying : router bgp 65502
Applying :   shutdown
Applying : router ospf p1
Applying :   shutdown
Applying : router ospfv3 p1
Applying :   shutdown
```

Maintenance mode operation successful.

This example shows how to automatically boot the switch into maintenance mode if a fatal error occurs:

```
switch(config)# system mode maintenance on-reload reset-reason fatal_error
```

Triggering Graceful Insertion

When you finish performing any debugging or upgrade operations, you can trigger a graceful insertion to restore all protocols.

Before you begin

If you want the system to use a normal-mode profile that you create, see [Configuring the Maintenance-Mode Profile, on page 509](#).

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	no system mode maintenance [dont-generate-profile] Example: <pre>switch(config)# no system mode maintenance dont-generate-profile Following configuration will be applied: no ip pim isolate router bgp 65502 no isolate router ospf p1 no isolate router ospfv3 p1 no isolate Do you want to continue (y/n)? [no] y Starting to apply commands... Applying : no ip pim isolate Applying : router bgp 65502 Applying : no isolate Applying : router ospf p1 Applying : no isolate Applying : router ospfv3 p1 Applying : no isolate Maintenance mode operation successful. Generating Current Snapshot</pre>	Puts all enabled protocols in normal mode (using the no isolate command). The dont-generate-profile option prevents the dynamic searching of enabled protocols and executes commands configured in a normal-mode profile. Use this option if you want the system to use a normal-mode profile that you have created. The system prompts you to continue. Enter y to continue or n to terminate the process.
Step 3	(Optional) show system mode Example: <pre>switch(config)# show system mode System Mode: Normal</pre>	Displays the current system mode. The switch is now in normal mode and is fully operational.

Maintenance Mode Enhancements

Starting with Release 7.0(3)I5(1), the following maintenance mode enhancements have been added to Cisco Nexus 9000 Series switches:

- In the system maintenance shutdown mode, the following message is added:

NOTE: The command `system interface shutdown` will shutdown all interfaces excluding `mgmt0`.

- Entering the CLI command, **system mode maintenance** checks and sends alerts for the orphan ports.
- In isolate mode, when the vPC is configured, the following message is added:

NOTE: If you have vPC orphan interfaces, please ensure `vpc orphan-port suspend` is configured under them, before proceeding further.

- Custom Profile Configuration: A new CLI command, **system mode maintenance always-use-custom-profile** is added for custom profile configuration. A new CLI command, **system mode maintenance non-interactive** is added for Cisco Nexus 9000 Series switches only. It provides a way to facilitate the transition to maintenance mode or normal mode without confirmation being done or each step being printed on the CLI session.

When a loopback interface is configured with an IP address on a device, and this device is advertised to a peer device, then the device (with the loopback interface) moves to maintenance mode. In such a case, use the custom maintenance profile when **system interface shutdown** is configured on the device.

When you create a custom profile (in maintenance or normal mode), it displays the following message:

Please use the command **system mode maintenance always-use-custom-profile** if you want to always use the custom profile.

- A delay has been added before the `after_maintenance` snapshot is taken. The **no system mode maintenance** command exits once all the configuration for the normal mode has been applied, the mode has been changed to normal mode, and a timer has been started to take the `after_maintenance` snapshot. Once the timer expires, the `after_maintenance` snapshot is taken in the background and a new warning syslog, `MODE_SNAPSHOT_DONE` is sent once the snapshot is complete.

The final output of the CLI command **no system mode maintenance** indicates when the `after_maintenance` snapshot is generated:

The `after_maintenance` snapshot will be generated in `<delay>` seconds. After that time, please use `show snapshots compare before_maintenance after_maintenance` to check the health of the system. The timer delay for the `after_maintenance` snapshot is defaulted to 120 seconds but it can be changed by a new configuration command.

The new configuration command to change the timer delay for the `after_maintenance` snapshot is **system mode maintenance snapshot-delay <seconds>**. This configuration overrides the default setting of 120 seconds to any value between 0 and 65535 and it is displayed in the ASCII configuration.

A new show command, **show maintenance snapshot-delay** has also been added to display the current snapshot-delay value. This new show command supports the XML output.

- A visible CLI indicator has been added to display when the system is in the maintenance mode, for example, `switch (maint-mode) #`.

- Support for the SNMP traps has been added when the device moves from the maintenance mode to the normal mode and vice-versa through CLI reload, or system reset. The **snmp-server enable traps mmode cseMaintModeChangeNotify** trap is added to enable changing to the maintenance mode trap notification. The **snmp-server enable traps mmode cseNormalModeChangeNotify** is added to enable changing to the normal mode trap notification. Both the traps are disabled by default.

Verifying the GIR Configuration

To display the GIR configuration, perform one of the following tasks:

Command	Purpose
show interface brief	Displays abbreviated interface information.
show maintenance on-reload reset-reasons	Displays the reset reasons for which the switch comes up in maintenance mode. For a description of the maintenance mode reset reasons, see Triggering Graceful Removal, on page 514 .
show maintenance profile [maintenance-mode normal-mode]	Displays the details of the maintenance-mode or normal-mode profile.
show maintenance timeout	Displays the maintenance-mode timeout period, after which the switch automatically returns to normal mode.
show {running-config startup-config} mmode [all]	Displays the maintenance-mode section of the running or startup configuration. The all option includes the default values.
show snapshots	Displays snapshots present on the switch.
show snapshots compare <i>snapshot-name-1</i> <i>snapshot-name-2</i> [summary ipv4routes ipv6routes]	Displays a comparison of two snapshots. The summary option displays just enough information to see the overall changes between the two snapshots. The ipv4routes and ipv6routes options display the changes in IPv4 and IPv6 routes between the two snapshots.
show snapshots dump <i>snapshot-name</i>	Displays the content of each file that was generated when the snapshot was taken.
show snapshots sections	Displays the user-specified snapshot sections.
show system mode	Displays the current system mode.

Configuration Examples for GIR

The **redistribute direct** configuration under Border Gateway Protocol (BGP) will attract traffic as the BGP isolate mode does not withdraw direct routes. This example shows how to use the **route-map** command to enable BGP to withdraw direct routes in isolate mode.

Policy Configuration

Use the **route-map my-rmap-deny** command in maintenance mode to exclude SVIs with a tag 200 configuration.

```
switch(config)# route-map my-rmap-deny deny 10
switch(config-route-map)# match tag 200
switch(config-route-map)# exit
switch(config)# route-map my-rmap-deny permit 20
```

Use the **route-map my-rmap-permit** command in normal mode to include SVIs with a tag 200 configuration.

```
switch(config)# route-map my-rmap-permit permit 10
switch(config-route-map)# match tag 200
switch(config-route-map)# exit
switch(config)# route-map my-rmap-permit permit 20
```

Virtual IP (vIP)/Switch Virtual Interface (SVI) Configuration

```
switch(config)# interface loopback 200
switch(config-if)# ip address 192.0.2.100/8 tag 200
switch(config)# interface vlan 2
switch(config-if)# ip address 192.0.2.108/8 tag 200
....
switch(config)# interface vlan 3
switch(config-if)# ip address 192.0.2.102/8 tag 200
```

BGP Configuration

```
switch(config)# feature bgp
switch(config)# router bgp 100
switch(config-router)# neighbor 192.0.2.100
....
```

Maintenance Mode Profile

```
switch# configure maintenance profile maintenance-mode
switch(config-mm-profile)# router bgp 200
switch(config-mm-profile-router)# address-family ipv4 unicast
switch(config-mm-profile-router-af)# redistribute direct route-map my-rmap-deny
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# sleep instance 1 10
```

Normal Mode Profile

```
switch# configure maintenance profile normal-mode
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# address-family ipv4 unicast
switch(config-mm-profile-router-af)# redistribute direct route-map my-rmap-permit
switch(config-mm-profile-router-af)# exit
switch(config-mm-profile)# sleep instance 1 20
```




CHAPTER 32

Performing Software Maintenance Upgrades

This chapter describes how to perform software maintenance upgrades (SMUs) on Cisco NX-OS devices.

This chapter includes the following sections:

- [About SMUs, on page 523](#)
- [Prerequisites for SMUs, on page 525](#)
- [Guidelines and Limitations for SMUs, on page 525](#)
- [Performing a Software Maintenance Upgrade for Cisco NX-OS, on page 526](#)
- [Performing a Software Maintenance Upgrade for Guest Shell Bash, on page 545](#)
- [Additional References, on page 546](#)

About SMUs

A software maintenance upgrade (SMU) is a package file that contains fixes for a specific defect. SMUs are created to respond to immediate issues and do not include new features. Typically, SMUs do not have a large impact on device operations. SMU versions are synchronized to the package major, minor, and maintenance versions they upgrade.

The effect of an SMU depends on its type:

- Process restart SMU-Causes a process or group of processes to restart on activation.
- Reload SMU-Causes a parallel reload of supervisors and line cards.

SMUs are not an alternative to maintenance releases. They provide a quick resolution of critical issues. All defects fixed by SMUs are integrated into the next maintenance releases of upcoming software trains, as applicable. SMUs also have the following considerations:

- SMUs are created for the following:
 - Critical SIR PSIRTs without a workaround or fix
 - Severity1 and Severity2 issues without a workaround or fix
- If a fix is already available in a maintenance release of the same software train or already released on a later long-lived release, no SMU is provided. You are encouraged to acquire the fix from the maintenance release.



Note Depending on the fix, in some cases it may not be possible to provide an SMU. In such cases, the only option is to upgrade to the next maintenance release when available.

For information on upgrading your device to a new feature or maintenance release, see the [Cisco Nexus 9000 Series NX-OS Software Upgrade and Downgrade Guide](#).

For information on Cisco NX-OS optionality feature, see the [Cisco Nexus 9000 Series NX-OS Software Upgrade and Downgrade Guide](#).



Note Activating an SMU does not cause any earlier SMUs, or the package to which the SMU applies, to be automatically deactivated.

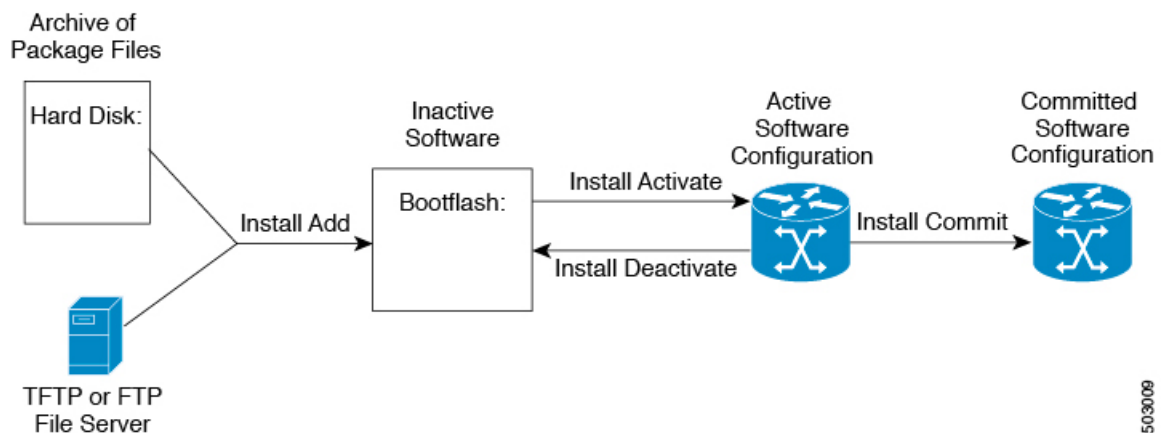
Package Management

The general procedure for adding and activating SMU packages on the device is as follows:

1. Copy the package file or files to a local storage device or file server.
2. Add the package or packages on the device using the **install add** command.
3. Activate the package or packages on the device using the **install activate** command.
4. Commit the current set of packages using the **install commit** command.
5. (Optional) Deactivate and remove the package.

The following figure illustrates the key steps in the package management process.

Figure 11: Process to Add, Activate, and Commit SMU Packages



Impact of Package Activation and Deactivation

The activation or deactivation of an SMU package can have an immediate impact on the system. The system can be affected in the following ways:

- New processes might be started.
- Running processes might be stopped or restarted.
- All processes in the line cards might be restarted. Restarting processes in the line cards is equivalent to a soft reset.
- The line cards might reload.
- No processes in the line cards might be affected.



Note You must address any issues that result from the revised configuration and reapply the configuration, if necessary.



Tip After the activation process completes, enter the **show install log** command to display the process results.

Prerequisites for SMUs

These prerequisites must be met for a package to be activated or deactivated:

- You must be in a user group associated with a task group that includes the proper task IDs. If you suspect a user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- Verify that all line cards are installed and operating properly. For example, do not activate or deactivate packages while line cards are booting, while line cards are being upgraded or replaced, or when you anticipate an automatic switchover activity.

Guidelines and Limitations for SMUs

SMUs have the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 9.3(9), you can apply a reload SMU along with (ND) ISSU to the same image version (currently running an image on the switch) without a disruptive reload of the switch. You can apply a reload SMU by upgrading to the same image version using ND-ISSU along with the reload SMU using the **install all nxos <same image> package <smu> non-disruptive** command.
- Beginning with Cisco NX-OS Release 10.3(4a)M, a new option is added to the **install all nxos** command, **keep-committed-smu**, to retain the previously installed SMU after upgrade to the same image version when the SMU is compatible with the target image. Execute the command as follows:

install all nxos <image> package <smu> keep-committed-smu

Example:

```
switch# install all nxos nxos64-cs.10.4.3.M.bin package
nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm keep-committed-smu non-disruptive
```

- No-reload options are supported in Cisco NX-OS Release 9.3(9) for the SMU installation. The **no-immediate-reload** option is used for activating or deactivating the SMU feature.
- Some packages require the activation or deactivation of other packages. If the SMUs have dependencies on each other, you cannot activate them without first activating the previous ones.
- SMU packages being activated must be compatible with the image version running in the switch.
- Activation is performed only after the package compatibility checks have been passed. If a conflict is found, an error message is displayed.
- You can install multiple SMU packages by creating a tar bundle. See the [Advanced SMU Installation Methods, on page 541](#) section for more details.
- While a software package is being activated, other requests are not allowed to run on any of the impacted nodes. Package activation is completed when a message similar to this one appears:

```
Install operation 1 completed successfully at Thu Jan 9 01:19:24 2014
```

- Each CLI install request is assigned a request ID, which can be used later to review the events.

Performing a Software Maintenance Upgrade for Cisco NX-OS

Preparing for Package Installation

You should use several **show** commands to gather information in preparation for the SMU package installation.

Before you begin

- Determine if a software change is required.
- Verify that the new package is supported on your system. Some software packages require that other packages or package versions be activated, and some packages support only specific line cards.
- Review the release notes for important information related to that release and to help determine the package compatibility with your device configuration.
- Verify that the system is up, stable, and prepared for the software changes.

Procedure

	Command or Action	Purpose
Step 1	show logging logfile grep -i "System ready" Example: <pre>switch# show logging logfile grep -i "System ready"</pre>	Displays if your system is up. Use this command to verify that the system is ready for SMU package installation. Configuring install commands before the system is ready, may result with an "Install operation 11 failed because cannot lock config" error message.

	Command or Action	Purpose
Step 2	show install active Example: <pre>switch# show install active</pre>	Displays the active software on the device. Use this command to determine what software should be added on the device and to compare to the active software report after installation operations are complete.
Step 3	show module Example: <pre>switch# show module</pre>	Confirms that all modules are in the stable state.
Step 4	show clock Example: <pre>switch# show clock</pre>	Verifies that the system clock is correct. Software operations use certificates based on device clock times.

Example

This example shows how to verify that the system is up. A "System ready" response indicates that the system is ready for SMU package installation.

```
switch# show logging logfile | grep -i "System ready"
2018 Feb 19 11:13:04 switch %ASCII-CFG-2-CONF_CONTROL: System ready
```

This example shows how to display the active packages for the entire system. Use this information to determine if a software change is required.

```
switch# show install active
Boot Image:
  NXOS Image: bootflash:///nxos.7.0.3.I7.3.1.bin

Active Packages:

switch#
```

This example shows how to display the current system clock setting:

```
switch# show clock
02:14:51.474 PST Wed Jan 04 2014
```

Downloading the SMU Package File from Cisco.com

Follow these steps to download the SMU package file:

Procedure

-
- Step 1** Log in to Cisco.com.
- Step 2** Go to the Download Software page at this URL: <http://software.cisco.com/download/navigator.html>

- Step 3** In the Select a Product list, choose **Switches > Data Center Switches > Cisco Nexus 9000 Series Switches > model**.
- Step 4** Choose the appropriate SMU file for your device and click **Download**.

Copying the Package File to a Local Storage Device or Network Server

You must copy the SMU package file to a local storage device or a network file server to which the device has access. After this task is done, the package can be added and activated on the device.

If you need to store package files on the device, we recommend that you store the files on the hard disk. The boot device is the local disk from which the package is added and activated. The default boot device is bootflash:.



Tip Before you copy package files to a local storage device, use the **dir** command to determine if the required package files are already on the device.

If the SMU package files are located on a remote TFTP, FTP, or SFTP server, you can copy the files to a local storage device. After the files are located on the local storage device, the package can be added and activated on the device from that storage device. The following server protocols are supported:

- Trivial File Transfer Protocol—TFTP allows files to be transferred from one computer to another over a network, usually without the use of client authentication (for example, username and password). It is a simplified version of FTP.



Note Some package files might be larger than 32 MB, and the TFTP services provided by some vendors might not support a file this large. If you do not have access to a TFTP server that supports files larger than 32 MB, download the file using FTP.

- File Transfer Protocol—FTP is part of the TCP/IP protocol stack and requires a username and password.
- SSH File Transfer Protocol—SFTP is part of the SSHv2 feature in the security package and provides for secure file transfers. For more information, see the [Cisco Nexus 9000 Series NX-OS Security Configuration Guide](#).



Note Consult your system administrator for the location and availability of your network server.

Use the commands in the following table to copy the SMU package file from the server to your device using the file transfer protocols.

Table 27: Commands for Copying SMU Package Files to the Device

Command	Purpose
copy tftp://hostname-or-ipaddress/directory-path/filename bootflash:	Copies the package file from the TFTP server to the bootflash:. • <i>hostname-or-ipaddress</i>—The hostname or IP address of the network file server. • <i>directory-path</i>—The network file server path that leads to the package file to be added. • <i>filename</i>—The name of the package file that you want to add.

Command	Purpose
copy ftp://username:password@hostname-or-ipaddress/directory-path/filename bootflash:	Copies the package file from the FTP server to the bootflash:. • <i>username</i>—The username of the user who has access privileges to the directory in which the package file is stored. • <i>password</i>—The password associated with the username of the user who has access privileges to the directory in which the package file is stored. If a password is not provided, the networking device accepts anonymous FTP. • <i>hostname-or-ipaddress</i>—The hostname or IP address of the network file server. • <i>directory-path</i>—The network file server path that leads to the package file to be added. The specified directory should be a directory under the home directory of the user. In this example, the file being downloaded is in a subdirectory called "images" in the home directory of the user "john." Note For FTP services, <i>directory-path</i> is the directory relative to the <i>username</i> home directory. If you want to specify an absolute path for the directory, you must add a "/" following the server address. • <i>filename</i>—The name of the package file that you want to add.

Command	Purpose
copy sftp://hostname-or-ipaddress/directory-path/filename bootflash:	Copies the package file from the SFTP server to the bootflash: • <i>hostname-or-ipaddress</i>—The hostname or IP address of the network file server. • <i>directory-path</i>—The network file server path that leads to the package file to be added. • <i>filename</i>—The name of the package file that you want to add.

After the SMU package file has been transferred to a network file server or the local storage device, you are ready to add and activate the file.

Adding and Activating Packages

You can add SMU package files that are stored on a local storage device or on a remote TFTP, FTP, or SFTP server to your device.



Note The SMU package being activated must be compatible with the currently active software to operate. When an activation is attempted, the system runs an automatic compatibility check to ensure that the package is compatible with the other active software on the device. If a conflict is found, an error message displays. The activation is performed only after all compatibility checks have been passed.



Note Activating an SMU does not cause any earlier SMUs or the package to which the SMU applies to be automatically deactivated.

Before you begin

Make sure that all packages to be added are present on a local storage device or a network file server.

Make sure that you meet all of the prerequisites for the activation of packages.

Complete the procedure described in [Copying the Package File to a Local Storage Device or Network Server](#), on page 528.

Procedure

	Command or Action	Purpose
Step 1	Connect to the console port and log in.	Establishes a CLI management session to the console port.

	Command or Action	Purpose
Step 2	(Optional) dir bootflash:	Displays the package files that are available to be added. Note Only SMU package files can be added and activated using this procedure.
Step 3	install add <i>filename</i> [activate] Example:	Unpacks the package software files from the local storage device or network server and adds them to the bootflash; and all active and standby supervisors installed on the device. The <i>filename</i> argument can take any of these formats: • bootflash:<i>filename</i> • tftp:<i>//hostname-or-ipaddress/directory-path/filename</i> • ftp:<i>//username:password@hostname-or-ipaddress/directory-path/filename</i> • usb1:<i>filename</i> • usb2:<i>filename</i> For all SMU packages except the CSCur02700 SMU package, you can use the optional activate keyword to automatically activate the package after it is added successfully. Note For the CSCur02700 SMU package, use the install activate command in Step 5 to activate the package. Do not use the optional activate keyword with the install add command as the package might fail and require a reboot. Multiple versions of an SMU package can be added to the storage device without impacting the running configuration, but only one version of a package can be activated for a line card. Note Press ? after a partial package name to display all possible matches available for activation. If there is only one match, press the Tab key to fill in the rest of the package name.
Step 4	(Optional) show install inactive Example: <code>switch# show install inactive</code>	Displays the inactive packages on the device. Verify that the package added in the previous step appears in the display.
Step 5	Required: install activate <i>filename</i> Example:	Activates a package that was added to the device. SMU packages remain inactive until activated. (Skip this step if the package was

	Command or Action	Purpose
	Example:	activated earlier with the install add activate command.) Tip After the activation process finishes, enter the show install log command to display the process results.
Step 6	Repeat Step 5 until all packages are activated.	Activates additional packages as required.
Step 7	(Optional) show install active Example: <code>switch# show install active</code>	Displays all active packages. Use this command to determine if the correct packages are active.

Committing the Active Package Set

When an SMU package is activated on the device, it becomes part of the current running configuration. To make the package activation persistent across system-wide reloads, you must commit the package on the device.



Note On startup, the device loads the committed package set. If the system is reloaded before the current active package is committed, the previously committed package set is used.

Before you begin

Before you commit a package set, verify that the device is operating correctly and is forwarding packets as expected.

Complete the procedure described in [Adding and Activating Packages, on page 531](#).

Procedure

	Command or Action	Purpose
Step 1	install commit <i>filename</i> Example:	Commits the current set of packages so that these packages are used if the device is restarted.
Step 2	(Optional) show install committed Example: <code>switch# show install committed</code>	Displays which packages are committed.

Example

This example shows how to commit active SMU packages on the device and then verify the committed packages:

Deactivating and Removing Packages

When a package is deactivated, it is no longer active on the device, but the package files remain on the boot disk. The package files can be reactivated later, or they can be removed from the disk.

The Cisco NX-OS software also provides the flexibility to roll back the selected package set to a previously saved package set. If you find that you prefer a previous package set over the currently active package set, you can use the **install deactivate** and **install commit** commands to make a previously active package set active again.

Before you begin

You cannot deactivate a package if it is required by another active package. When you attempt to deactivate a package, the system runs an automatic check to ensure that the package is not required by other active packages. The deactivation is performed only after all compatibility checks have been passed.

You cannot delete a package if it is part of the running or committed software of the device.

Procedure

	Command or Action	Purpose
Step 1	Connect to the console port and log in.	Establishes a CLI management session to the console port.
Step 2	install deactivate <i>filename</i> Example:	Deactivates a package that was added to the device and turns off the package features for the line card. Note You must run install commit after install deactivate to deactivate the package completely, otherwise the package gets activated again after reload. For reload SMU, run install commit after the device reloads.
Step 3	(Optional) show install inactive Example: switch# show install inactive	Displays the inactive packages on the device.
Step 4	(Optional) install commit Example: switch# install commit	Commits the current set of packages so that these packages are used if the device is restarted. Note

	Command or Action	Purpose
		Packages can be removed only if the deactivation operation is committed.
Step 5	(Optional) install remove <i>{filename inactive}</i> Example: Example: <pre>switch# install remove inactive Proceed with removing? (y/n)? [n] y</pre>	Removes the inactive package. • Only inactive packages can be removed. • Packages can be removed only if they are deactivated from all line cards in the device. • The package deactivation must be committed. • To remove a specific inactive package from a storage device, use the install remove command with the <i>filename</i> argument. • To remove all inactive packages from all nodes in the system, use the install remove command with the inactive keyword.

No-Reload Options for SMU Installation

The following are the no-reload options for SMU installation:

Method 1: CLI Install Add/Activate

```
switch# show version internal build-identifier
nxos image file: bootflash:///nxos64.10.2.0.184.bin : S184
switch# show install inactive
Boot Image:
  NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Inactive Packages:

Inactive Base Packages:
  tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
  tor-2.0.0.0-10.2.0.184.lib32_n9000
  tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
switch#
switch# install add nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm
[#####] 100%
Install operation 3 completed successfully at Mon Jul 12 11:32:28 2021

switch# show install inactive
Boot Image:
  NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Inactive Packages:
  nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 available

Inactive Base Packages:
  tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
  tor-2.0.0.0-10.2.0.184.lib32_n9000
  tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
switch#
```

```

switch# show install pkg-info nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000
Request timeout:: Success
Name       : nxos64.CSCaa12345-n9k_ALL
Version    : 1.0.0
Release    : 10.2.1
License    : Cisco proprietary
Patch Type : reload
Requires   : core
Provides   : nxos64.CSCaa12345-n9k_ALL
Conflicts  :
Description : This is a patch for CSCaa12345-n9k_ALL
switch#

```

CLI Install Activate PATCH with no-immediate-reload option

```

switch# install activate nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 ?
<CR>
WORD                Package Name
forced              Non-interactive
no-immediate-reload Skip immediate reload for reload type patches.
switch# install activate nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000
no-immediate-reload
[#####] 100%
Install operation 4 !!WARNING!! This patch will get activated only after
a reload of the switch. at Mon Jul 12 11:33:50 2021

switch#
switch# show install inactive
Boot Image:
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin

Inactive Packages:
    nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 activate_pending_reload

Inactive Base Packages:
    tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
    tor-2.0.0.0-10.2.0.184.lib32_n9000
    tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000

switch#
switch# show install patch
Boot Image:
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin

-----
nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 Inactive Committed
(activate_pending_reload)
-----

switch##

switch# reload
This command will reboot the system. (y/n)? [n] y

CISCO SWITCH Ver7.69
Switch G2
Device detected on 0:1:2 after 0 msec
Device detected on 0:1:1 after 0 msec
Device detected on 0:1:0 after 0 msec
....

After switch reload, wait for system in ready state

:///nxos64.10.2.0.184.bin : S184
switch#

switch# show logging logfile | include ready

```

```
2021 Jul 12 11:40:34 N93180-1 %ASCII-CFG-2-CONF_CONTROL: System ready
```

```
switch#
```

```
switch# show install patch
```

```
Boot Image:
```

```
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin
```

```
-----
nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 Active
-----
```

```
switch#
```

```
switch# show install active
```

```
Boot Image:
```

```
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin
```

```
Active Packages:
```

```
    nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 active
```

```
Active Base Packages:
```

```
....
```

CLI Install Deactivate PATCH with no-immediate-reload option

```
switch# install deactivate nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 ?
<CR>
```

```
WORD
```

```
Package Name[Note: startup configuration may get affected]
```

```
forced
```

```
Non-interactive
```

```
no-immediate-reload Skip immediate reload for reload type patches.
```

```
switch# install deactivate nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000
```

```
no-immediate-reload
```

```
[#####] 100%
```

```
Install operation 5 !!WARNING!! This patch will get deactivated only after
a reload of the switch. at Mon Jul 12 11:42:24 2021
```

```
switch#
```

```
switch# show install patch
```

```
Boot Image:
```

```
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin
```

```
-----
nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 Active (deactivate_pending_reload)
-----
```

```
switch#
```

```
switch# show install active
```

```
Boot Image:
```

```
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin
```

```
Active Packages:
```

```
    nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 active
```

```
Active Base Packages:
```

```
....
```

```
switch# reload
```

```
WARNING: Uncommitted patches present
```

```
This command will reboot the system. (y/n)? [n] y
```

```
CISCO SWITCH Ver7.69
```

```
Switch G2
```

```
Device detected on 0:1:2 after 0 msec
```

```
Device detected on 0:1:1 after 0 msecs
Device detected on 0:1:0 after 0 msecs
....
```

After switch reload, wait for system in ready state

```
switch# show logging logfile | include ready
2021 Jul 12 11:52:28 N93180-1 %ASCII-CFG-2-CONF_CONTROL: System ready
switch#
```

```
switch# show install patch
Boot Image:
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin
```

```
-----
nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 Inactive Committed
-----
```

```
switch# show install inactive
Boot Image:
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin
```

```
Inactive Packages:
    nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000 available
```

```
Inactive Base Packages:
    tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
    tor-2.0.0.0-10.2.0.184.lib32_n9000
    tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
```

```
switch#
```

```
switch# install remove nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000
Proceed with removing nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000? (y/n)? [n] y
[#####] 100%
Install operation 6 completed successfully at Mon Jul 12 11:57:06 2021
switch# show install patch
Boot Image:
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin
```

```
-----
switch# show install inactive
Boot Image:
    NXOS Image: bootflash:///nxos64.10.2.0.184.bin
```

```
Inactive Packages:
```

```
Inactive Base Packages:
    tahusd_common-1.0.0.0-10.2.0.184.lib32_64_n9000
    tor-2.0.0.0-10.2.0.184.lib32_n9000
    tor_n9k-2.0.0.0-10.2.0.184.lib32_n9000
```

```
switch#
```

CLI Install ADD ACTIVATE via bootflash: with no-immediate-reload

```
switch# install add nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm activate ?
<CR>
downgrade          Downgrade package
forced             Non-interactive
no-immediate-reload Skip immediate reload for reload type patches.
upgrade            Upgrade package
```

```
switch# install add nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm activate
```



```

no-immediate-reload
Adding the patch (/nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm)
[#####] 100%
Install operation 7 completed successfully at Mon Jul 12 12:03:02 2021

Activating the patch (/nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm)
[#####] 100%
Install operation 8 !!WARNING!! This patch will get activated only after
a reload of the switch. at Mon Jul 12 12:03:10 2021

switch#

```

CLI Install ADD ACTIVATE via tftp with no-immediate-reload

```

switch# install add
tftp://172.27.250.42/auto/tftp-sjc-users1/shuojiun/nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm
vrf management activate ?
<CR>
downgrade          Downgrade package
forced             Non-interactive
no-immediate-reload Skip immediate reload for reload type patches.
upgrade            Upgrade package

switch# install add
tftp://172.27.250.42/auto/tftp-sjc-user1/tester/nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm
vrf management activate no-immediate-reload
[#####] 100%
Install operation 11 !!WARNING!! This patch will get activated only after
a reload of the switch. at Mon Jul 12 12:06:49 2021

switch#

```

Method 2: VIA DME REST Action/Exec payload



Note In the payload below, "reloadFlag": "noreload", you need to set "reloadFlag" as "noreload". "reloadFlag" is not new in the Action/Exec items.

```

POST URL:
http://172.27.250.239//api/mo/sys/action.json

{
  "actionLCont": {
    "children": [
      {
        "actionLSubj": {
          "attributes": {
            "dn": "sys/action/lsubj-[sys]"
          },
          "children": [
            {
              "topSystemSwpkgsInstallTask": {
                "attributes": {
                  "dn":
"sys/action/lsubj-[sys]/topSystemSwpkgsInstallTask",
                  "pkgAction": "add-activate",
                  "reloadFlag": "noreload",
                  "adminSt": "start",
                  "url":
"nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000.rpm"
                }
              }
            }
          ]
        }
      }
    ]
  }
}

```

```

    ]
  }
}

{
  "actionLCont": {
    "children": [
      {
        "actionLSubj": {
          "attributes": {
            "dn": "sys/action/lsubj-[sys]"
          },
          "children" : [
            {
              "topSystemSwpkgsInstallTask": {
                "attributes": {
                  "dn":
"sys/action/lsubj-[sys]/topSystemSwpkgsInstallTask",
                  "pkgAction": "activate",
                  "reloadFlag": "noreload",
                  "adminSt": "start",
                  "url":
"nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000"
                }
              }
            }
          ]
        }
      }
    ]
  }
}

{
  "actionLCont": {
    "children": [
      {
        "actionLSubj": {
          "attributes": {
            "dn": "sys/action/lsubj-[sys]"
          },
          "children" : [
            {
              "topSystemSwpkgsInstallTask": {
                "attributes": {
                  "dn":
"sys/action/lsubj-[sys]/topSystemSwpkgsInstallTask",
                  "pkgAction": "deactivate",
                  "reloadFlag": "noreload",
                  "adminSt": "start",
                  "url":
"nxos64.CSCaa12345-n9k_ALL-1.0.0-10.2.1.lib32_64_n9000"
                }
              }
            }
          ]
        }
      }
    ]
  }
}

```

```
}
}
```

Advanced SMU Installation Methods

Installing Multiple SMU Packages Using a Single TAR File

If you want to install multiple SMU packages, you can create a single TAR bundle file and use it across the switches in the Data Center.

Follow these steps to generate a TAR file from a given list of SMU packages downloaded from a software download center.



Note

The file name mentioned in the following examples is for illustration purposes only, and the actual file name will depend on the appropriate release.

Procedure

Step 1 Create a new folder in the user computer or virtual machine.

```
bash# mkdir nx1043
```

Step 2 Download the required SMU packages from the Cisco Software Download Center portal and copy the SMU packages to the new folder.

```
bash# cp nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm nx1043/
bash# cp nxos64-cs.CSCxy22222-1.0.0-10.4.3.lib32_64_n9000.rpm nx1043/
```

Step 3 Create a tar bundle file.

```
bash# cd nx1043
bash# tar cf nxos64-cs.10.4.3.smu.bundle.tar *.rpm
```

Step 4 Use the existing **install add filename activate** command to install the SMU packages from the TAR bundle.

```
switch# install add nxos64-cs.10.4.3.smu.bundle.tar activate
```

Installing SMU Packages as Part of the New NX-OS Software Image Installation

On Cisco Nexus switches, the NX-OS software image can be upgraded to a newer version using the **install all** command. This command has been enhanced to include SMU packages apart from the NX-OS switch software image, which benefits the software maintenance operations by reducing the number of reload required during the installation process for both the software image and SMU packages.

The **install all** command can be initiated with a single .tar bundle file containing either:

- One NX-OS software image and a single SMU .rpm file
- One NX-OS software image and a tar bundle of multiple SMU .rpm files



Note The child tar bundle must not contain a mix of SMU .rpm files and another tar bundle of SMU .rpm files.

When the **install all** command is initiated with one or more SMU .rpm files, the switch will automatically commit the SMU files after the upgrade.

If the switch is reloaded during bootup, the SMUs will not be applied and will remain in an inactive state. The SMUs can be installed using the **install all** or **install activate** commands.

The following section describes all the supported scenarios when SMU packages are included in the **install all** command.



Note The filename mentioned in the following examples is for illustration purposes only, and the actual filename will depend on the appropriate release.

Example-1: In this scenario, a new software image and a single SMU package is used.

```
switch# install all nxos nxos64-cs.10.4.3.M.bin package
nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm
```

Example-2: In this scenario, a set of SMU packages are created as a TAR bundle following the TAR file method mentioned above and installed along with the NX-OS software image.

```
switch# install all nxos nxos64-cs.10.4.3.M.bin package nxos64-cs.10.4.3.smu.bundle.tar
```

Example-3: In this scenario, a single SMU package and the NX-OS software image can be bundled into one single tar file and installed using the **install all** command.

```
switch# install all nxos nxos64-cs.10.4.3.M.SMU.plus.IMAGE.tar
```

1. Download the SMU package from the Cisco Download center. For example:
nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm
2. Download **nxos64-cs.10.4.3.M.bin** and place it in the same folder.
3. Create a tar bundle **nxos64-cs.10.4.3.M.SMU.plus.IMAGE.tar** consisting of the NX-OS image and SMU package.

```
bash# tar cf nxos64-cs.10.4.3.M.SMU.plus.IMAGE.tar nxos64-cs.10.4.3.M.bin
nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm
```

Example-4: When there are multiple SMU packages must be installed along with the NX-OS image, the SMU packages must be built into a SMU tar bundle file first as explained in the [Installing Multiple SMU Packages Using a Single TAR File, on page 541](#) section. Subsequently, this SMU tar bundle can be further bundled together with the NX-OS image and a single tar file could be used in the **install all** command.

```
Switch# install all nxos nxos64-cs.10.4.3.M.SMU.BUNDLE.plus.IMAGE.tar
```

1. Create a SMU tar bundle image with the list of SMU packages as explained in the [Installing Multiple SMU Packages Using a Single TAR File, on page 541](#) section.

```
bash# mkdir nx1043
bash# cp nxos64-cs.CSCxy11111-1.0.0-10.4.3.lib32_64_n9000.rpm nx1043/
bash# cp nxos64-cs.CSCxy22222-1.0.0-10.4.3.lib32_64_n9000.rpm nx1043/
bash# cd nx1043
bash# tar cf nxos64-cs.10.4.3.smu.bundle.tar *.rpm
```

2. Download **nxos64-cs.10.4.3.M.bin** and place it in the same folder.

3. Create another tar bundle **nxos64-cs.10.4.3.M.SMU.BUNDLE.plus.IMAGE.tar**.

```
bash# tar cf nxos64-cs.10.4.3.M.SMU.BUNDLE.plus.IMAGE.tar nxos64-cs.10.4.3.M.bin
nxos64-cs.10.4.3.smu.bundle.tar
```

Downgrading Feature RPMs

Follow this procedure to downgrade an installed feature RPM to the base feature RPM.

Procedure

	Command or Action	Purpose
Step 1	(Optional) show install packages Example: switch# show install packages ntp.lib32_n9000 1.0.1-7.0.3.I2.2e installed	Displays the feature RPM packages on the device.
Step 2	Required: run bash Example: switch# run bash bash-4.2\$	Loads Bash.
Step 3	Required: cd /rpms Example: bash-4.2\$ cd /rpms	Changes to the RPMs folder in Bash.
Step 4	Required: ls *feature* Example: bash-4.2\$ ls *ntp* ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm	Lists the RPM for the specified feature.
Step 5	Required: cp filename /bootflash Example: bash-4.2\$ cp ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm /bootflash	Copies the base feature RPM to the bootflash.
Step 6	Required: exit Example: bash-4.2\$ exit	Exits Bash.
Step 7	Required: install add bootflash:filename activate downgrade Example: switch# install add bootflash:ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm activate downgrade	Downgrades the feature RPM. Note If you are prompted to reload the device, enter Y . A reload is required only when downgrading the NTP and SNMP feature RPMs.

	Command or Action	Purpose
	<pre> Adding the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) [#####] 60% Adding the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) [#####] 100% Install operation 11 completed successfully at Thu Sep 8 15:35:35 2015 Activating the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) This install operation requires system reload. Do you wish to continue (y/n)?: [n] y [217.975959] [1473348971] writing reset reason 132, System reset due to reload patch(es) activation [217.991166] [1473348971]\ufffd\ufffd CISCO SWITCH Ver7.51 Device detected on 0:6:0 after 0 msecs Device detected on 0:1:1 after 0 msecs Device detected on 0:1:0 after 0 msecs MCFrequency 1333Mhz Relocated to memory </pre>	
Step 8	(Optional) show install packages i feature Example: <pre> switch# show install packages i ntp ntp.lib32_n9000 1.0.0-7.0.3.I2.2e installed </pre>	Displays the base feature RPM on the device.

Displaying Installation Log Information

The installation log provides information on the history of the installation operations. Each time an installation operation is run, a number is assigned to that operation.

- Use the **show install log** command to display information about both successful and failed installation operations.
- Use the **show install log** command with no arguments to display a summary of all installation operations. Specify the *request-id* argument to display information specific to an operation. Use the **detail** keyword to display details for a specific operation, including file changes, nodes that could not be reloaded, and any impact to processes.

This example shows how to display information for all installation requests:

This example shows how to display additional information, including any impact to nodes and processes:

This example shows the output after an SMU package has been activated but before the switch has been reloaded:

Performing a Software Maintenance Upgrade for Guest Shell Bash

You can perform a software maintenance upgrade for Bash in the Guest Shell.

Procedure

	Command or Action	Purpose
Step 1	Download the SMU package file for Guest Shell Bash from Cisco.com.	Obtains the package file from Cisco.com. For instructions, see Downloading the SMU Package File from Cisco.com , on page 527.
Step 2	Copy the SMU package file to the bootflash: of the switch.	Copies the package file to the device. For instructions, see Copying the Package File to a Local Storage Device or Network Server , on page 528.
Step 3	guestshell Example: switch# guestshell guestshell:~\$	Accesses the Guest Shell.
Step 4	sudo rpm -Uvh /bootflash/filename Example: guestshell:~\$ sudo rpm -Uvh /bootflash/bash-4.2-r8.x86_64.rpm Preparing... ##### [100%] 1: bash ##### [100%] update-alternatives: Linking //bin/sh to /bin/bash	Upgrades the existing Bash file in the Guest Shell.
Step 5	rpm -qa grep bash Example: guestshell:~\$ rpm -qa grep bash bash-4.2-r8.x86_64	Verifies that the new version of the Bash file was installed successfully.
Step 6	guestshell sync Example: switch# guestshell sync Access to the guest shell will be temporarily disabled while it synchronizes contents to standby. Are you sure you want to continue? (y/n) [n] y	On a dual-supervisor system, synchronizes the rootfs with the Bash SMU version to the standby supervisor before doing a switchover. If you do not run this command, you will need to repeat this procedure after a supervisor switchover. Note

	Command or Action	Purpose
	<pre>dt-n9k3-1# 2014 Oct 7 05:00:01 dt-n9k3-1 %\$ VDC-1 %\$ %VMAN-2-INSTALL_STATE: Deactivating virtual service 'guestshell+' dt-n9k3-1# 2014 Oct 7 05:00:06 dt-n9k3-1 %\$ VDC-1 %\$ %VMAN-2-ACTIVATION_STATE: Successfully deactivated virtual service 'guestshell+' 2014 Oct 7 05:00:12 dt-n9k3-1 %\$ VDC-1 %\$ %VMAN-2-ACTIVATION_STATE: Successfully deactivated virtual service 'guestshell+' ; Starting sync to standby sup 2014 Oct 7 05:00:32 dt-n9k3-1 %\$ VDC-1 %\$ %VMAN-2-MOVE_STATE: Successfully synced virtual service 'guestshell+' ; Activating 2014 Oct 7 05:00:32 dt-n9k3-1 %\$ VDC-1 %\$ %VMAN-2-ACTIVATION_STATE: Activating virtual service 'guestshell+' 2014 Oct 7 05:00:56 dt-n9k3-1 %\$ VDC-1 %\$ %VMAN-2-ACTIVATION_STATE: Successfully activated virtual service 'guestshell+'</pre>	The new Bash file is preserved after a Guest Shell reboot or Guest Shell disable+enable. However, you need to reinstall the Guest Shell Bash SMU package file after a Guest Shell destroy+enable.

Additional References

Related Documents

Related Topic	Document Title
Software upgrades	<i>Cisco Nexus 9000 Series NX-OS Software Upgrade and Downgrade Guide</i>



CHAPTER 33

Performing Configuration Replace

This chapter includes the following sections:

- [About Configuration Replace and Commit-timeout, on page 547](#)
- [Overview, on page 547](#)
- [Guidelines and Limitations for Configuration Replace, on page 549](#)
- [Recommended Workflow for Configuration Replace, on page 551](#)
- [Performing a Configuration Replace, on page 552](#)
- [Verifying Configuration Replace, on page 554](#)
- [Examples for Configuration Replace, on page 555](#)

About Configuration Replace and Commit-timeout

The configuration replace feature enables you to replace the running configuration of the Cisco Nexus switch with the user provided configuration without reloading the device. The device reload may be required only when a configuration itself requires a reload. The running configuration file that is provided by the user should be taken using copy running file. Unlike **copy file: to running**, the configuration replace feature is not a merge operation. This feature replaces the entire running configuration with a new configuration that is provided by the user. If there is a failure in the configuration replace, the original configuration is restored in the switch. From Cisco NX-OS Release 9.3(1), **best-effort** option is introduced. This option enables the configuration replace to execute the full patch despite any error in the commands and the original configuration is not restored in the switch.

The commit-timeout feature enables you to rollback to the previous configuration after successfully performing the configuration replace operation. If the commit timer expires, the rollback operation is automatically initiated.



Note

- You must provide a valid running configuration that has been received with the Cisco NX-OS device. It should not be a partial configuration.

Overview

The configuration replace feature has the following operation steps:

- Configuration replace intelligently calculates the difference between the current running-configuration and the user-provided configuration in the Cisco Nexus switch and generates a patch file which is the difference between the two files. You can view this patch file which includes a set of configuration commands.
- Configuration replace applies the configuration commands from the patch file similarly to executing commands.
- The configuration rolls back to or restores the previous running configuration under the following situations:
 - If there is a mismatch in the configuration after the patch file has been applied.
 - If you perform the configuration operation with a commit timeout and the commit timer expires.
- The configuration does not roll back to or does not restore the previous running configuration when the best-effort option is used. This option enables the configuration replace to execute the full patch despite any error in the commands and will not roll back to the previous configuration.
- You can view the exact configuration that caused a failure using the **show config-replace log exec** command.
- Restore operations that fail while restoring the switch to the original configuration, are not interrupted. The restore operation continues with the remaining configuration. Use the **show config-replace log exec** command to list the commands that failed during the restore operation.
- If you enter the **configure replace commit** command before the timer expires, the commit timer stops and the switch runs on the user provided configuration that has been applied through the configuration replace feature.
- If the commit timer expires, roll back to the previous configuration is initiated automatically.
- In Cisco NX-OS Release 9.3(1), semantic validation support is added for the configuration replace. This semantic validation is done as part of the precheck in configuration replace. The patch gets applied only when the semantic validation is successful. After applying the patch file, configuration replace triggers the verification process. The configuration replace compares the running-configuration with the user configuration file during the verification process. If there is a mismatch, it restores the device to the original configuration.

The differences between configuration replace and copying a file to the running-configuration are as follows:

Configuration Replace	Copying a file
The configure replace <i><target-url></i> command removes the commands from the current running-configuration that are not present in the replacement file. It also adds commands that need to be added to the current running-configuration.	The copy <i><source-url></i> running-config command is a merge operation which preserves all the commands from, both the source file and the current running-configuration. This command does not remove the commands from the current running-configuration that are not present in the source file.
You must use a complete Cisco NX-OS configuration file as the replacement file for the configure replace <i><target-url></i> command.	You can use a partial configuration file as a source file for the copy <i><source-url></i> running-config command.

Benefits of Configuration Replace

The benefits of configuration replace are:

- You can replace the current running-configuration file with the user-provided configuration file without having to reload the switch or manually undo CLI changes to the running-configuration file. As a result, the system downtime is reduced.
- You can revert to the saved Cisco NX-OS configuration state.
- It simplifies the configuration changes by allowing you to apply a complete configuration file to the device, where only the commands that need to be added or removed are affected. The other service and configurations that are not modified remain untouched.
- If you configure the commit-timeout feature, you can rollback to the previous configuration even when the configuration replace operation has been successful.

Guidelines and Limitations for Configuration Replace

The configuration replace feature has the following configuration guidelines and limitations:

- The configuration replace feature is supported on Cisco Nexus 3000 Series and Cisco Nexus 9000 Series switches.
- Only one user can perform the configuration replace, checkpoint, and rollback operations, or copy the running-configuration to the startup configuration at the same time. Parallel operations such as operations via multiple Telnet, SSH, or NX-API sessions are not supported. The multiple configuration replace or rollback request is serialized, for example, only after the first request is completed, processing of the second request begins.
- You are not allowed to initiate another configuration replace operation when the commit timer is running. You must either stop the timer by using the **configure replace commit** command or wait until the commit timer expires before you initiate another configuration replace operation.
- When **system default switchport shutdown** or **no system default switchport shutdown** is used with **configure replace bootflash:target_config_file** command, the user should make sure that desired port state (shutdown or no shutdown) statement is present in the target_config_file for all switchport interfaces.
- For a successful configuration replace operation, sequence number must be present for all ACE entries in ACL in the target configuration file.
- Beginning with Cisco NX-OS Release 9.3(6), the **boot nxos image** configuration can be excluded in the **show running-config**, **show startup-config**, **copy running-config filename**, and **copy startup-config filename** commands by configuring **service exclude-bootconfig**.
- The commit-timeout feature is initiated only if you perform the configuration replace operation with the commit-timeout. The timer value range is from 30 to 3600 seconds.
- The user provided configuration file must be the valid show running-configuration output that is taken from the Cisco NX-OS device (copy run file). The configuration cannot be a partial configuration and must include mandated commands, such as user admin and so on.

- We do not recommend a configuration replace operation that is performed on the configuration file that is generated across the software version because this operation could fail. A new configuration file must be regenerated whenever there is change in the software version.
- The configuration replace operation is not supported if you attempt to replace a multichassis EtherChannel trunk (MCT) configuration with a virtual peer-link configuration. This operation is not allowed because the physical MCT uses the CFS distribution over Ethernet mode and the virtual peer-link use the CFS distribution over IP mode.
- We recommend that you do not change any configuration from others sessions if the configuration replace operation is in progress because it could cause the operation to fail.
- Note the following about the configuration replace feature:
 - Beginning with Cisco NX-OS Release 9.3(5), configuration replace (CR) for FEX interface configurations is supported. Provisioning of FEX is not supported through CR. Once provisioned, configurations on the FEX interfaces can modified through CR.
 - The configuration replace feature does not work if the FEX line card is offline.
 - The configuration replace feature is not supported on Cisco Nexus 9500 platform switches with -R line cards.
 - Beginning with Cisco NX-OS Release 9.3(5), the configuration replace feature is supported on port profiles.
 - The configuration replace feature is supported **only** for the configure terminal mode commands. The configure profile, configure jobs, and any other modes are not supported.
 - Beginning with Cisco NX-OS Release 9.3(5), the configure jobs mode is supported. Configuration files with scheduler job commands can be used for configuration replace.
 - Beginning with Cisco NX-OS Release 9.3(4), the configuration replace feature is supported for breakout interface configurations.
 - The configuration replace feature could fail if the running configuration includes the **feature-set mpls** or the **mpls static range** commands and tries to move to a configuration without MPLS or modifies the label range.
 - The configuration replace feature does not support autoconfigurations.
- If the line card to which the configuration replace feature is applied is offline, the configuration replace operation fails.
- An ITD service must be shut down (**shutdown**) prior to making ITD changes with the configuration replace feature.
- Entering maintenance mode from the user configuration is not supported.
- Using the **configure replace** command from maintenance mode asks for a user-confirmation with the following warning:


```
Warning: System is in maintenance mode. Please ensure user config won't inadvertently
revert back config in maintenance mode profile.
Do you wish to proceed anyway? (y/n) [n]
```
- Using the **configure replace** command from maintenance mode with a *<non-interactive>* option is supported. It takes the yes user-confirmation by default and proceeds.

- Sequence number is mandatory for CLI **ip community-list** and **ip as-path access-list** commands. Without a sequence number, the configuration replace operation fails.
- If your configurations demand reloading the Cisco NX-OS device in order to apply the configuration, then you must reload these configurations after the configuration replace operation.
- The order of the commands in the user provided configuration file must be the same as those commands in the running configuration of the Cisco Nexus switch.
- The user configuration file to which you need to replace the running configuration on the switch using CR should be generated from the running-config of the switch after configuring the new commands. The user configuration file should not be manually edited with the CLI commands and the sequence of the configuration commands should not be altered.
- The semantic validation is not supported in 4-Gig memory platforms.
- When different versions of a feature are present in the running configuration and user configuration (for example: VRRPv2 and VRRPv3), semantic validation option does not work as expected. This issue is a known limitation.
- In "verify-only" mode, the TCAM-dependent configuration may not throw an error and gets succeeded. However, it may fail during actual CR operation. To avoid this, it is recommended to apply TCAM carving configuration and reload before performing CR.
- Beginning from Cisco NX-OS Release 10.3(1)F, the configuration replace feature does not support feature app-hosting.

Recommended Workflow for Configuration Replace

The following workflow is the recommended workflow for configuration replace:



Note

- This workflow needs to be the same in the candidate config.
 - Default configuration in the candidate config is not supported.
-

1. Generate a configuration file by first applying the configurations on a Cisco Nexus Series device and then use the **show running-configuration** output as the configuration file. Use this file to make configuration modifications as required. Then use this generated or updated configuration file to perform configuration replace.



Note

Whenever there is a change in the software version, regenerate the configuration file. Do not use a configuration file, which is generated across different software versions, for the configuration replace operation.

2. View and verify the patch file by executing the **configure replace <file> show-patch** command. This is an optional step.
3. Run the configuration replace file either using or skipping the **commit-timeout <time>** feature. Based on your requirements, you can perform one of the following steps:

- Run **configure replace** *<file>* **verbose** to see the commands that get executed with configuration replace on the console.
 - Run the **configure replace** [**bootflash/scp/sftp**] *<user-configuration-file>* **verbose** **commit-timeout** *<time>* commands to configure the commit time.
4. Run the **configure replace commit** command to stop the commit timer. This step is necessary if you have run the configuration replace operation with the commit-timeout feature.
 5. Configuration replace performs a precheck that includes the semantic validation of the configuration. The configuration replace operation fails if there is an error. Use the **show config-replace log verify** command to see the details of the failed configurations. After applying the patch file, configuration replace triggers the verification process. The configuration replace compares the running-configuration with the user configuration file during the verification process. If there is a mismatch, it restores the device to the original configuration. Use the **show config-replace log verify** command to see the mismatched configurations.
 6. You can perform the following configuration replace operations in Cisco NX-OS Release 9.3(1):
 - Configuration replace without the semantic validation and without best-effort mode.
 - Configuration replace without the semantic validation and with best-effort mode.
 - Configuration replace with the semantic validation and without best-effort mode.
 - Configuration replace with the semantic validation and with best-effort mode.

Performing a Configuration Replace

To perform configuration replace, do the following:

Before you begin

Ensure that there is no conflict in ip address in the current configuration and candidate configuration files. An example for conflict in ip address is—consider that you configured 172.16.0.1/24 on eth interface 1/53 on the current configuration file and port channel 30 with 172.16.0.1/24 and 192.168.0.1/24 on eth 1/53 in the candidate configuration file. When you perform a configuration replace to the candidate configuration file, this results in a conflict in ip address.

Procedure

	Command or Action	Purpose
Step 1	configure replace { <i><uri_local></i> <i><uri_remote></i> } [verbose show-patch]	Performs configuration replace. If you make the configuration changes through any sessions when configuration replace is in progress, the configuration replace operation fails. If you send a configuration replace request when one configuration request is already in progress, then it gets serialized.

	Command or Action	Purpose
Step 2	configure replace [bootflash / scp / sftp] < <i>user-configuration-file</i> > show-patch	Displays the differences between the running-configuration and the user-provided configuration. Note • This command does not encrypt plain text password. • This command can still show a patch even after configuration replace is successful for CLI snmp-server traps commands.
Step 3	configure replace [bootflash / scp / sftp] < <i>user-configuration-file</i> > verbose	Replaces the configuration on the switch with the new user configuration that is provided by the user. Configuration replace is always atomic.
Step 4	configure replace < <i>user-configuration-file</i> > [best-effort]	Replaces the configuration on the switch with the new user configuration and enables the configuration replace with semantic validation. The best-effort option enables the configuration replace to execute the full patch despite any error in the commands and also make sure that the previous configuration is not rolled back. Beginning with Cisco NX-OS Release 10.5(1)F, configuration replace feature supports batch ACL configurations on Cisco Nexus 9300-FX2/FX3/GX Series switches. If the best effort mode is enabled, any failure within the batched configuration will result in skipping the entire set of configurations in that particular batch.
Step 5	configure replace < <i>user-configuration-file</i> > [verify-and-commit]	Replaces the configuration on the switch with the new user configuration and enables the configuration replace with semantic validation. The verify-and-commit option is used for enabling the semantic validation. Patch will be executed only if semantic validation of the full patch gets passed. You can use the best-effort option or the verify-and-commit option or both the options at the same time.

	Command or Action	Purpose
Step 6	configure replace <user-configuration-file> [verify-only]	Shows only the patch and does Semantic validation on the patch, and display the results. The patch does not get applied to the system.
Step 7	(Optional) configure replace [bootflash / scp / sftp] <user-configuration-file> verbose commit-timeout <time>	Configures the commit time in seconds. The timer starts after the configuration replace operation is successfully completed.
Step 8	(Optional) configure replace [commit]	Stops the commit timer and continues the configuration replace configuration. Note This step is applicable only if you have configured the commit-timeout feature. Note To rollback to the previous configuration, you must wait for the expiry of the commit timer. Once the timer expires, the switch is automatically rolled back to the previous configuration.
Step 9	(Optional) configure replace [bootflash/scp/sftp] <user-configuration-file> <i>non-interactive</i>	There is no user prompt in maintenance mode. The yes user-confirmation is taken by default, and rollback proceeds. You can use the non-interactive option only in the maintenance mode.
Step 10	(Optional) configure replace <user-configuration-file> [on-failure reload [save-startup-config]]	This option reloads the switch to the previously saved startup configuration if configure replace operation fails and the restore to the previous running configuration also fails. This option has no impact if: • Configure replace is successful • Configure replace fails, but restore to the previous running configuration is successful Note [save-startup-config] – This option saves the running config to startup before starting the configure replace operations.

Verifying Configuration Replace

To check and verify configuration replace and its status, use the commands that are outlined in the table:

Table 28: Verifying Configuration Replace

Command	Purpose
configure replace [bootflash/scp/sftp]<user-configuration-file> show-patch	Displays the difference between the running-configurations and user-provided configurations.
show config-replace log exec	Displays a log of all the configurations executed and those that failed. In case of an error, it displays an error message against that configuration.
show config-replace log verify	Displays the configurations that failed, along with an error message. It does not display configurations that were successful.
show config-replace status	Displays the status of the configuration replace operations, including in-progress, successful, and failure. If you have configured the commit-timeout feature, the commit and timer status and the commit timeout time remaining is also displayed.

Examples for Configuration Replace

See the following configuration examples for configuration replace:

- Use the **configure replace bootflash: <file> show-patch** CLI command to display the difference between the running-configurations and user-provided configurations.

```
switch(config)# configure replace bootflash:<file> show-patch
Collecting Running-Config
Converting to checkpoint file
#Generating Rollback Patch
!!
no role name abc
```

- Use the **configure replace bootflash: <file> verbose** CLI command to replace the entire running-configuration in the switch with the user-configuration.

```
switch(config)# configure replace bootflash:<file> verbose
Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallely may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
=====
config t
no role name abc
=====
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

Sample Example with adding of BGP configurations.
```

```

switch(config)# sh run | section bgp
switch(config)# sh file bootflash:file | section bgp
feature bgp
router bgp 1
  address-family ipv4 unicast
  neighbor 1.1.1.1
switch(config)#
switch(config)# configure replace bootflash:file verbose
Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallelly may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
=====
config t
feature bgp
router bgp 1
address-family ipv4 unicast
neighbor 1.1.1.1
=====
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

switch(config)# sh run | section bgp
feature bgp
router bgp 1
  address-family ipv4 unicast
  neighbor 1.1.1.1

Sample Example with ACL
switch(config)# configure replace bootflash:run_1.txt
Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallelly may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
=====
config t
no ip access-list nexus-50-new-xyz
ip access-list nexus-50-new-xyz-jkl-abc
10 remark Newark
20 permit ip 17.31.5.0/28 any
30 permit ip 17.34.146.193/32 any
40 permit ip 17.128.199.0/27 any
50 permit ip 17.150.128.0/22 any
=====
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

switch(config)#

switch(config)# show run aclmgr | sec nexus-50-new-xyz-jkl-abc
ip access-list nexus-50-new-xyz-jkl-abc
  10 remark Newark
  20 permit ip 17.31.5.0/28 any

```

```
30 permit ip 17.34.146.193/32 any
40 permit ip 17.128.199.0/27 any
50 permit ip 17.150.128.0/22 any
```

- Use the **configure replace bootflash:user-config.cfg verify-only** CLI command to generate and verify the patch semantically.

```
switch(config)# configure replace bootflash:user-config.cfg verify-only

Version match between user file and running configuration.
Pre-check for User config PASSED
Collecting Running-Config
Converting to checkpoint file
Generating Rollback Patch
Validating Patch
=====
`config t `
`interface Ethernet1/1`
`shutdown`
`no switchport trunk allowed vlan`
`no switchport mode`
`no switchport`
`exit`
Skip non dme command for CR validation
`interface Vlan1`
`shutdown`
`interface Ethernet1/1`
`shutdown`
`no switchport`
`ip address 1.1.1.1/24`
`exit`
Skip non dme command for CR validation
=====
Patch validation completed successful
switch(config)#
```

- Use the **configure replace bootflash:user-config.cfg best-effort verify-and-commit** CLI command to replace the switch running configuration with the given user configuration after performing the semantic validation on patch.

```
switch(config)# configure replace bootflash:user-config.cfg best-effort verify-and-commit

Version match between user file and running configuration.
Pre-check for User config PASSED
ADVISORY: Config Replace operation started...
Modifying running configuration from another VSH terminal in parallel
is not recommended, as this may lead to Config Replace failure.

Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Collecting Running-Config
Generating Rollback Patch

Validating Patch
Patch validation completed successful
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Rollback Patch

Configure replace completed successfully. Please run 'show config-replace log exec' to
```

see if there is any configuration that requires reload to take effect.

```
switch(config)#
```

- Use the **show config-replace log exec** CLI command to check all the configuration that is executed and failures if any.

```
switch(config)# show config-replace log exec
Operation          : Rollback to Checkpoint File
Checkpoint file name : .replace_tmp_28081
Scheme             : tmp
Rollback done By    : admin
Rollback mode       : atomic
Verbose             : enabled
Start Time          : Wed, 06:39:34 25 Jan 2017
```

```
-----
time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS
End Time          : Wed, 06:39:47 25 Jan 2017
Rollback Status    : Success
```

```
Executing Patch:
-----
switch#config t
switch#no role name abc
```

- Use the **show config-replace log verify** CLI command to check the failed configuration if any.

```
switch(config)# show config-replace log verify
Operation          : Rollback to Checkpoint File
Checkpoint file name : .replace_tmp_28081
Scheme             : tmp
Rollback done By    : admin
Rollback mode       : atomic
Verbose             : enabled
Start Time          : Wed, 06:39:34 25 Jan 2017
End Time            : Wed, 06:39:47 25 Jan 2017
Status              : Success
```

```
Verification patch contains the following commands:
-----
!!
! No changes
-----
```

```
time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS
```

- Use the **show config-replace status** CLI command to check the status of configuration replace.

```
switch(config)# show config-replace status
Last operation : Rollback to file
Details:
  Rollback type: atomic replace_tmp_28081
  Start Time: Wed Jan 25 06:39:28 2017
  End Time: Wed Jan 25 06:39:47 2017
  Operation Status: Success
switch(config)#
```

Configure Replace might fail when the manually created configuration is used instead of the configuration generated from the switch. The reason for possible failures is the potential difference in the default configuration that isn't shown in the show running configuration. Refer to the following examples:

If the power redundant command is the default command, it doesn't get displayed in the default configuration. But it's displayed when you use the **show run all** command. See the following example:

```
switch# show run all

!Command: show running-config all
!Running configuration last done at: Tue Nov 12 11:07:44 2019
!Time: Tue Nov 12 11:16:09 2019

version 9.3(1) Bios:version 05.39
power redundancy-mode ps-redundant
no hardware module boot-order reverse
no license grace-period
<snip>
hostname n9k13
```

The power redundant command isn't shown in the show running configuration command out. See the following example:

```
!Command: show running-config
!Running configuration last done at: Tue Nov 12 11:07:44 2019
!Time: Tue Nov 12 11:17:24 2019

version 9.3(1) Bios:version 05.39
hostname n9k13
```

When the **power redundancy-mode ps-redundant** command is added in the user configuration for the configure replace; then the verification/commit might fail. See the following example:

```
switch# show file bootflash:test

!Command: show running-config
!Running configuration last done at: Tue Nov 12 10:56:49 2019
!Time: Tue Nov 12 11:04:57 2019

version 9.3(1) Bios:version 05.39
power redundancy-mode ps-redundant
hostname n9k13
```

The **power redundancy-mode ps-redundant** command will not be shown in the show running after configure replace; therefore it will be considered as “missing” and the CR will fail. An example is given below.

```
switch# config replace bootflash:test verify-and-commit

Version match between user file and running configuration.
Pre-check for User config PASSED
ADVISORY: Config Replace operation started...
Modifying running configuration from another VSH terminal in parallel
is not recommended, as this may lead to Config Replace failure.

Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Collecting Running-Config
.Generating Rollback Patch

Validating Patch
Patch validation completed successful
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Rollback Patch
Executing Rollback Patch
During CR operation,will retain L3 configuration
```

```

when vrf member change on interface
Generating Running-config for verification
Generating Patch for verification
Verification failed, Rolling back to previous configuration
Collecting Running-Config
Cleaning up switch-profile buffer
Generating Rollback patch for switch profile
Executing Rollback patch for switch profiles. WARNING - This will change the
configuration of switch profiles and will also affect any peers if configured
Collecting Running-Config
Generating Rollback Patch
Rollback Patch is Empty
Rolling back to previous configuration is successful

Configure replace failed. Use 'show config-replace log verify' or 'show config-replace
log exec' to see reasons for failure

n9k13# show config-replace log verify
Operation : Config-replace to user config
Checkpoint file name : .replace_tmp_31849
Scheme : tmp
Cfg-replace done By : agargula
Cfg-replace mode : atomic
Verbose : disabled
Start Time : Tue, 11:20:59 12 Nov 2019
Start Time UTC : Tue, 10:20:59 12 Nov 2019
-----
End Time : Tue, 11:21:28 12 Nov 2019
End Time UTC : Tue, 10:21:28 12 Nov 2019
Status : Failed

Verification patch contains the following commands:
-----
!!
Configuration To Be Added Missing in Running-config
=====
!
power redundancy-mode ps-redundant

Undo Log
-----
End Time : Tue, 11:21:32 12 Nov 2019
End Time UTC : Tue, 10:21:32 12 Nov 2019
Status : Success
n9k13#

```

In the above example, CR will consider the default commands that are missing and will therefore fail.



CHAPTER 34

Configuring Rollback

This chapter describes how to configure rollback on Cisco NX-OS devices.

This chapter contains the following sections:

- [About Rollbacks, on page 561](#)
- [Prerequisites for Rollbacks, on page 562](#)
- [Guidelines and Limitations for Rollbacks, on page 562](#)
- [Default Settings for Rollbacks, on page 563](#)
- [Configuring Rollbacks, on page 563](#)
- [Verifying the Rollback Configuration, on page 565](#)
- [Configuration Example for Rollback, on page 565](#)
- [Additional References, on page 566](#)

About Rollbacks

A rollback allows you to take a snapshot, or user checkpoint, of the Cisco NX-OS configuration and then reapply that configuration to your device at any point without having to reload the device. A rollback allows any authorized administrator to apply this checkpoint configuration without requiring expert knowledge of the features configured in the checkpoint.

Cisco NX-OS automatically creates system checkpoints. You can use either a user or system checkpoint to perform a rollback.

You can create a checkpoint copy of the current running configuration at any time. Cisco NX-OS saves this checkpoint as an ASCII file which you can use to roll back the running configuration to the checkpoint configuration at a future time. You can create multiple checkpoints to save different versions of your running configuration.

When you roll back the running configuration, you can trigger the following rollback types:

- **atomic**—Implement a rollback only if no errors occur.
- **best-effort**—Implement a rollback and skip any errors.
- **stop-at-first-failure**—Implement a rollback that stops if an error occurs.

The default rollback type is atomic.

When you are ready to roll back to a checkpoint configuration, you can view the changes that will be applied to your current running configuration before committing to the rollback operation. If an error occurs during the rollback operation, you can choose to cancel the operation, or ignore the error and proceed with the rollback.

If you cancel the operation, Cisco NX-OS provides a list of changes already applied before the error occurred. You need to clean up these changes manually.

Automatically Generated System Checkpoints

The Cisco NX-OS software automatically generates system checkpoints to help you avoid a loss of configuration information. System checkpoints are generated by the following events:

- Disabling an enabled feature with the **no feature** command
- Removing an instance of a Layer 3 protocol, such as with the **no router bgp** command or the **no ip pim sparse-mode** command
- License expiration of a feature

If one of these events causes system configuration changes, the feature software creates a system checkpoint that you can use to roll back to the previous system configuration. The system generated checkpoint filenames begin with “system-” and include the feature name. For example, the first time that you disable the EIGRP feature, the system creates the checkpoint named system-fm-__inst_1__eigrp.

High Availability

Whenever a checkpoint is created using the `checkpoint` or `checkpoint checkpoint_name` commands, the checkpoint is synchronized to the standby unit.

A rollback remembers the states of the checkpoint operation, so if the checkpoint operation is interrupted and the system is left in an inconsistent state, a rollback can complete the checkpoint operation (synchronize the checkpoint with the standby unit) before proceeding with the rollback operation.

Your checkpoint files are still available after a process restart or supervisor switchover. Even if there is an interruption during the process restart or supervisor switchover, the checkpoint will complete successfully before proceeding with the operation. In a supervisor switchover, the checkpoint is completed on the new active unit.

If a process restart or supervisor switchover occurs during a rollback operation, after the restart or switchover completes, the rollback will resume from its previous state and complete successfully.

Virtualization Support

Cisco NX-OS creates a checkpoint of the running configuration. You can create different checkpoint copies.

Prerequisites for Rollbacks

To configure rollback, you must have network-admin user privileges.

Guidelines and Limitations for Rollbacks

Rollbacks have the following configuration guidelines and limitations:

- You can create up to ten checkpoint copies.

- Your checkpoint filenames must be 80 characters or less.
- You cannot start a checkpoint filename with the word *system*.
- You can start a checkpoint filename with the word *auto*.
- You can name a checkpoint file *summary* or any abbreviation of the word *summary*.
- Only one user can perform a checkpoint, rollback, or copy the running configuration to the startup configuration at the same time.
- After the system executes the **write erase** and **reload** command, checkpoints are deleted. You can use the **clear checkpoint database** command to clear out all checkpoint files.
- Although a rollback is not supported for checkpoints across software versions, users can perform a rollback at their own discretion and can use the best-effort mode to recover from errors.
- When checkpoints are created on bootflash, differences with the running-system configuration cannot be performed before performing the rollback, and the system reports “No Changes.”
- Checkpoints created using the **checkpoint** and **checkpoint checkpoint_name** commands are present upon a switchover.
- Checkpoints are present upon reload unless a **write-erase** command is issued before a reload.
- A rollback to files on bootflash is supported only on files that are created using the **checkpoint checkpoint_name** command and not on any other type of ASCII file.
- Checkpoint names must be unique. You cannot overwrite previously saved checkpoints with the same name.
- Rollback is not supported in the context of auto configurations. Checkpoints do not store auto configurations. Therefore, after a rollback is performed, the corresponding auto configurations will not be present.
- Multiple port VLAN mappings configured on an interface during a rollback operation cause the rollback feature to fail.

Default Settings for Rollbacks

This table lists the default settings for rollback parameters.

Parameters	Default
Rollback type	Atomic

Configuring Rollbacks



Note Be aware that the Cisco NX-OS commands may differ from the Cisco IOS commands.

Creating a Checkpoint

You can create up to ten checkpoints of your configuration.

Procedure

	Command or Action	Purpose
Step 1	[no] checkpoint {[<i>cp-name</i>] [description <i>descr</i>] file <i>file-name</i> } Example: <pre>switch# checkpoint stable</pre>	Creates a checkpoint of the running configuration to either a user checkpoint name or a file. The checkpoint name can be any alphanumeric string up to 80 characters but cannot contain spaces. If you do not provide a name, Cisco NX-OS sets the checkpoint name to <i>user-checkpoint-number</i> where <i>number</i> is from 1 to 10. The description can contain up to 80 alphanumeric characters, including spaces. You can use the no form of the checkpoint command to remove a checkpoint name. Use the delete command to remove a checkpoint file.
Step 2	(Optional) show checkpoint <i>cp-name</i> [all] Example: <pre>switch# show checkpoint stable</pre>	Displays the contents of the checkpoint name.

Implementing a Rollback

You can implement a rollback to a checkpoint name or file. Before you implement a rollback, you can view the differences between source and destination checkpoints that reference current or saved configurations.



Note If you make a configuration change during an atomic rollback, the rollback will fail.

Procedure

	Command or Action	Purpose
Step 1	show diff rollback-patch {checkpoint <i>src-cp-name</i> running-config startup-config file <i>source-file</i>} {checkpoint <i>dest-cp-name</i> running-config startup-config file <i>dest-file</i>} Example: <pre>switch# show diff rollback-patch checkpoint stable running-config</pre>	Displays the differences between the source and destination checkpoint selections.

	Command or Action	Purpose
Step 2	rollback running-config {checkpoint <i>cp-name</i> file <i>cp-file</i>} [atomic best-effort stop-at-first-failure] Example: <pre>switch# rollback running-config checkpoint stable</pre>	Creates a rollback to the specified checkpoint name or file. You can implement the following rollback types: • atomic—Implement a rollback only if no errors occur. • best-effort—Implement a rollback and skip any errors. • stop-at-first-failure—Implement a rollback that stops if an error occurs. The default is atomic. This example shows how to implement a rollback to a user checkpoint name.

Verifying the Rollback Configuration

To display the rollback configuration information, perform one of the following tasks:

Command	Purpose
show checkpoint <i>name</i> [all]	Displays the contents of the checkpoint name.
show checkpoint all [user system]	Displays the contents of all checkpoints. You can limit the displayed checkpoints to user or system generated checkpoints.
show checkpoint summary [user system]	Displays a list of all checkpoints. You can limit the displayed checkpoints to user or system generated checkpoints.
show diff rollback-patch {checkpoint <i>src-cp-name</i> running-config startup-config file <i>source-file</i>} {checkpoint <i>dest-cp-name</i> running-config startup-config file <i>dest-file</i>}	Displays the differences between the source and destination checkpoint selections.
show rollback log [exec verify]	Displays the contents of the rollback log.

Use the **clear checkpoint database** command to delete all checkpoint files.

Configuration Example for Rollback

This example shows how to create a checkpoint file and then implements a best-effort rollback to a user checkpoint name:

```
checkpoint stable
rollback running-config checkpoint stable best-effort
```

Additional References

Related Documents

Related Topic	Document Title
Configuration files	<i>Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide</i>



CHAPTER 35

Integrity Check of Candidate Config

This chapter describes how to perform integrity check of Candidate Config.

This chapter includes the following sections:

- [About Candidate Config, on page 567](#)
- [Guidelines and Limitations for Candidate Config Integrity Check, on page 567](#)
- [Performing Integrity Check for Candidate Config, on page 568](#)
- [Examples of Integrity Check, on page 568](#)

About Candidate Config

Candidate config is a subset of the running-config which checks whether the Candidate config exists in the running-config without any additions or modifications or deletions.

To check the integrity of the candidate config, use the following commands:

- `show diff running-config`
- `show diff startup-config`

For more information on the CLIs, refer to [Performing Integrity Check for Candidate Config, on page 568](#).

Guidelines and Limitations for Candidate Config Integrity Check

Candidate config integrity check has the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 10.2(3)F, Candidate config integrity check option is introduced on all Cisco Nexus switches.
- If you must perform an integrity check on a full running configuration as input instead of a partial config, then it is recommended not to use the **partial** keyword.
- The line numbers that are displayed in the generated running config do not match with the candidate config as they are internally generated one.
- If there is any difference between the configuration of running and candidate, then it is displayed inline as output.

- If the whole block of configuration in the candidate file is a new addition, it will be appended at the end of the generated running config.
- When the candidate config has an SNMP or an AAA user CLI with clear-text password, the SNMP user is seen as a diff even when the user is already configured.

Performing Integrity Check for Candidate Config

To perform the integrity check, use the following commands:

Before you begin



Note Before performing the integrity check, ensure that the running config and the candidate config belong to the same image version.

Procedure

	Command or Action	Purpose
Step 1	show diff running-config <i>file_url</i> [unified] Example: <pre>switch# show diff running-config bootflash:candidate.cfg partial unified</pre>	Displays the differences between the running and user given candidate config. • <i>file_url</i>: File path to compare with. • unified: Displays the difference between running and user configuration in unified format.
Step 2	show diff startup-config <i>file_url</i> [unified] Example: <pre>switch# show diff startup-config bootflash:candidate.cfg unified</pre>	Displays the differences between the startup and user given candidate config. • <i>file_url</i>: File path to compare with. • unified: Displays the difference between startup and user configuration in unified format.

Examples of Integrity Check

No Difference Between Running and Candidate Config

```
switch# show diff running-config bootflash:base_running.cfg
switch#
```

Difference Between Running and Candidate

```
switch# show diff running-config bootflash:modified-running.cfg unified
--- running-config
+++ User-config
@@ -32,11 +32,11 @@

interface Ethernet1/1
    mtu 9100
    link debounce time 0
    beacon
-   ip address 2.2.2.2/24
+   ip address 1.1.1.1/24
    no shutdown

interface Ethernet1/2

interface Ethernet1/3
switch#
```

Difference Between Running and Partial Candidate

```
switch# show file bootflash:intf_vlan.cfg
interface Vlan101
    no shutdown
    no ip redirects
    ip address 1.1.2.1/24 secondary
    ip address 1.1.1.1/24
switch#
switch# show diff running-config bootflash:intf_vlan.cfg partial unified
--- running-config
+++ User-config
@@ -3897,10 +3883,14 @@
    mtu 9100
    ip access-group IPV4_EDGE in
    ip address 2.2.2.12/26 tag 54321

    interface Vlan101
+   no shutdown
+   no ip redirects
+   ip address 1.1.2.1/24 secondary
+   ip address 1.1.1.1/24

    interface Vlan102
        description Vlan102
        no shutdown
        mtu 9100
switch#
```




CHAPTER 36

Performing Secure Erase

- [Information about Secure Erase, on page 571](#)
- [Prerequisites for Performing Secure Erase, on page 571](#)
- [Guidelines and Limitations for Secure Erase, on page 572](#)
- [Configuring Secure Erase, on page 572](#)

Information about Secure Erase

Beginning with Cisco NX-OS Release 10.2(2)F, the Secure Erase feature is introduced to erase all customer information for Nexus 9000 switches. Secure Erase is an operation to remove all the identifiable customer information on Cisco NX-OS devices in conditions of product removal due to Return Merchandise Authorization (RMA), or upgrade or replacement, or system end-of-life.

Cisco Nexus 9000 switches consume storage to conserve system software images, switch configuration, software logs, and operational history. These areas can have customer-specific information such as details regarding network architecture, and design as well as a potential target for data thefts.

The Secure Erase process is used in the following two scenarios:

- Return Material Authorization (RMA) for a device - If you must return a device to Cisco for RMA, remove all the customer-specific data before obtaining an RMA certificate for the device.
- Recovering the compromised device - If the key material or credentials that are stored on a device is compromised, reset the device to factory configuration, and then reconfigure the device.



Note Secure Erase feature will not erase content in External storage.

The device reloads to perform a factory reset which results in the EoR chassis modules to enter the power down mode. After a factory reset, the device clears all configuration, logs, and storage information.

Prerequisites for Performing Secure Erase

- Ensure that all the software images, configurations, and personal data are backed up before performing the secure erase operation.

- Ensure that there is an uninterrupted power supply when the process is in progress.
- Ensure that neither In-Service Software Upgrade (ISSU) nor In-Service Software Downgrade (ISSD) is in progress before starting the secure erase process.

Guidelines and Limitations for Secure Erase

- FX3 or FX3S or FX3P switches are supported in TOR and FEX mode. If secure erase is done in FEX mode, a switch will boot in TOR mode after the secure erase operation.
- Software patches, if installed on the device, will not be restored after the Secure Erase operation.
- If the **factory-reset** command is issued through a session, the session is not restored after the completion of the factory reset process.

The top of rack switches and supervisor modules returns to the loader prompt.

End of row switch modules will be in a powered down state.

If you configure secure erase of fex, the factory reset is initiated and fex configuration will be removed.

Fex secure erase to be monitored using fex console. In case of failure, reboot and bring up fex and initiate secure erase again.

Configuring Secure Erase

To delete all necessary data before shipping to RMA, configure secure erase using the below command:

Command	Purpose
factory-reset modulemod Example: <pre>switch(config)# factory-reset [module <3>]</pre>	Use the command with all options enabled. No system configuration is required to use the factory reset command. To secure erase for fex, use factory-resetfex [<i>allfex_no</i>] - To secure erase all fex at once, use option all. Note Ensure that the fex is not in Active-Active scenario, before initiating secure erase operation. Use the option mod to reset the start-up configurations: - For top of rack switches, the command is factory-reset or factory-reset module 1. - In LXC mode for top of rack switches, the command is factory-reset module 1 or 27 - For end of row module switches, the command is [{module <module> [bypass-secure-erase] [preserve-image]}] Beginning with Cisco NX-OS Release 10.2(3), the following options are supported for the factory-reset command: bypass-secure-erase: Use this option when secure data removal is not required (repartition and reformat storage only). preserve-image: This option preserves the running image and autoboots after the completion of erase operations. After the factory reset process is successfully completed, the switch reboots and is powered down.



Note Parallel secure erase operations are not supported. To erase more than one module in single EoR chassis, the recommended order is line card, fabric, standby supervisor, system controller, and then active supervisor.

You can boot that secure erase image to trigger the data wipe.

The following is an example output for configuring secure erase factory reset command:

```
FX2-2- switch#
FX2-2- switch# show fex
FEX          FEX          FEX          FEX
Number      Description    State        Model
```

```

Serial
-----
109          FEX0109          Online          N2K-C2348TQ-10GE
FOC1816R0F2
110          FEX0110          Online          N2K-C2348TQ-10G-E
FOC2003R1SQ

FX2-2-switch# factory-reset fex all
!!!! WARNING:
This command will perform factory-reset of all FEX modules !!!!
The factory reset operation will erase ALL persistent storage on the specified FEX module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable. Please, proceed with
caution and understanding that this operation cannot be undone and will leave the system
in a fresh-from-factory state.
!!!! WARNING !!!!

Do you want to continue? (y/n) [n] y
Initiating factory-reset for the FEX: 109 --- SUCCESS!!
FEX: 109 is reloading for the reset operation to proceed.
Factory reset may take time...
Please, wait and do not power off the FEX...
Trying to remove the FEX:109 config !!!
Initiating factory-reset for the FEX: 110 --- SUCCESS!!
FEX: 110 is reloading for the reset operation to proceed.
Factory reset may take time...
Please, wait and do not power off the FEX...
Trying to remove the FEX:110 config !!!
Successfully removed FEX:110 config. !!!

```

The following shows the example of fex logs:

```

FX2-2-switch# 2021
FEX console logs:
=====
bgl-ads-4157:138> telnet 10.127.118.15 2007
Trying 10.127.118.15...
Connected to 10.127.118.15.
Escape character is '^]'.

fex-109#
fex-109# [129266.313614] writing reset reason 9, Factory-reset requested by abc
[129266.391801] Restarting system - Factory-reset requested by abc [9]
U-Boot 2011.12 (Jun 25 2014 - 16:28:41) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Golden image
U-boot retry count 0
Jump to upgradeable image at 0xefd20040
U-Boot 2011.12 (Jun 25 2014 - 16:19:54) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz

```

```
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Upgradeable image
DRAM: Configuring DDR for 666.667 MT/s data rate
Time-out count = 480
DDR configuration get done
1 GiB (DDR3, 32-bit, CL=6, ECC on)
Memory test from 0x40000 to 0x1fdffff
Data line test..... OK
Address line test..... OK
OK
Flash: 288 MiB
L2: 256 KB enabled
Set dbglevel to its default value (0x1)
PCIE1: Root Complex of mini PCIe SLOT, x1, regs @ 0xffe0a000
PCIE1: Bus 00 - 01
PCIE2: Root Complex of PCIe SLOT, no link, regs @ 0xffe09000
PCIE2: Bus 02 - 02
Net: eTSEC1, eTSEC3
Hit Ctrl-L to stop autoboot: 0
WARN: user forced bootcmd="run sysboot"
.. WARNING: adjusting available memory to 30000000
## Booting kernel from Legacy Image at 01000000 ...
Image Name: Linux-2.6.27.47
Created: 2015-11-20 10:22:39 UTC
Image Type: PowerPC Linux Kernel Image (gzip compressed)
Data Size: 8936305 Bytes = 8.5 MiB
Load Address: 00000000
Entry Point: 00000000
Verifying Checksum ... OK
## Flattened Device Tree blob at 00c00000
Booting using the fdt blob at 0x00c00000
Uncompressing Kernel Image ... OK
Loading Device Tree to 03ffb000, end 03ffff82 ... OK
setup_arch: bootmem
mpc85xx_fex_setup_arch()
arch: exit
[0.436112] Host controller irq 17
[0.477490] pci 0000:00:00.0: ignoring class b20 (doesn't match header type 01)
[0.566841] Assign root port irq 17 for 0000:00:00.0
[2.210329] Enabling all PCI devices
[2.802226] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[2.975494] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[3.889037]
[3.889041] Watchdog init<0>
Mount failed for selinuxfs on /selinux: No such file or directory
INIT: version 2.86 booting
Setting system clock: [ OK ]
Mounting all filesystems: [ OK ]
/sbin/dhclient-script: configuration for eth1 not found. Continuing with defaults.
/etc/sysconfig/network-scripts/network-functions: line 78: eth1: No such file or directory
Mounting system image: [ OK ]
Unpacking system image: [ OK ]
```

```

Uncompressing system image: [ OK ]
Loading system image: [ OK ]
net.ipv4.ip_forward = 0
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_no_pmtu_disc = 1
Starting internet superserver: inetd [ OK ]
net.core.rmem_max = 524288
net.core.wmem_max = 524288
net.core.rmem_default = 524288
net.core.wmem_default = 524288
net.core.somaxconn = 1024
net.core.netdev_max_backlog = 1024
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[23.255118] Device eth0 configured with sgmi interface
Non issu restart
[24.151321]
[24.151327] base_addr is 26524<0>
Secure erase requested! Please, do not power off module!
Starting the secure erase. !!
This may take time. Please wait !!
>>>> Wiping all storage devices ...
[28.706882] NX-OS starts punching watchdog
grep: Backu: No such file or directory
+++ Starting mtd secure erase for the partition /dev/mtd2 +++
Erasing /dev/mtd2 ...
Erasing 128 Kibyte @ 17e0000 -- 99 % complete.
---> SUCCESS
Writing random data onto /dev/mtd2
Filling /dev/mtd2 using random data ...
Erasing blocks: 192/192 (100%)
Writing data: 24576k/24576k (100%)
Verifying data: 24576k/24576k (100%)
---> SUCCESS
Erasing /dev/mtd2 ...
Erasing 128 Kibyte @ 17e0000 -- 99 % complete.
---> SUCCESS
+++ Skipping cmos secure erase +++
>>>> Done
+++ Skipping nvram secure erase +++
>>>> Done
>>>> Iniatilzing system to factory defaults ...
+++ Starting init-system +++
Initializing /dev/mtd5
/isan/bin/mount_jffs2.sh: line 68: ${LOG_FILE}: ambiguous [ 651.954326] Restarting system.
U-Boot 2011.12 (Jun 25 2014 - 16:28:41) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Golden image
U-boot retry count 1
Jump to upgradeable image at 0xefd20040
U-Boot 2011.12 (Jun 25 2014 - 16:19:54) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,

```

```

CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Upgradeable image
DRAM: Configuring DDR for 666.667 MT/s data rate
Time-out count = 480
DDR configuration get done
1 GiB (DDR3, 32-bit, CL=6, ECC on)
Memory test from 0x40000 to 0x1fdffff
Data line test..... OK
Address line test..... OK
OK
Flash: 288 MiB
L2: 256 KB enabled
Set dbglevel to its default value (0x1)
PCIE1: Root Complex of mini PCIe SLOT, x1, regs @ 0xffe0a000
PCIE1: Bus 00 - 01
PCIE2: Root Complex of PCIe SLOT, no link, regs @ 0xffe09000
PCIE2: Bus 02 - 02
Net: eTSEC1, eTSEC3
Hit Ctrl-L to stop autoboot: 0
WARN: user forced bootcmd="run sysboot"
.. WARNING: adjusting available memory to 30000000
## Booting kernel from Legacy Image at 01000000 ...
Image Name: Linux-2.6.27.47
Created: 2015-11-20 10:22:39 UTC
Image Type: PowerPC Linux Kernel Image (gzip compressed)
Data Size: 8936305 Bytes = 8.5 MiB
Load Address: 00000000
Entry Point: 00000000
Verifying Checksum ... OK
## Flattened Device Tree blob at 00c00000
Booting using the fdt blob at 0x00c00000
Uncompressing Kernel Image ... OK
Loading Device Tree to 03ffb000, end 03fffe82 ... OK
setup_arch: bootmem
mpc85xx_fex_setup_arch()
arch: exit
[ 0.436112] Host controller irq 17
[ 0.477490] pci 0000:00:00.0: ignoring class b20 (doesn't match header type 01)
[ 0.566841] Assign root port irq 17 for 0000:00:00.0
[ 2.210556] Enabling all PCI devices
[ 2.804559] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[ 2.975502] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[ 3.889014]
[ 3.889018] Watchdog init<0>
Mount failed for selinuxfs on /selinux: No such file or directory
INIT: version 2.86 booting
Setting system clock: [ OK ]
Mounting all filesystems: [ OK ]
/sbin/dhclient-script: configuration for eth1 not found. Continuing with defaults.
/etc/sysconfig/network-scripts/network-functions: line 78: eth1: No such file or directory

```

```

Mounting system image: [ OK ]
Unpacking system image: [ OK ]
Uncompressing system image: [ OK ]
Loading system image: [ OK ]
net.ipv4.ip_forward = 0
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_no_pmtu_disc = 1
Starting internet superserver: inetd [ OK ]
net.core.rmem_max = 524288
net.core.wmem_max = 524288
net.core.rmem_default = 524288
net.core.wmem_default = 524288
net.core.somaxconn = 1024
net.core.netdev_max_backlog = 1024
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[ 22.630994] Device eth0 configured with sgmi interface
Non issu restart
[ 23.535827]
[ 23.535832] base_addr is 26524<0>
INIT: Entering runlevel: 3
fex login: Sorry, user root is not allowed to execute '/sbin/sysctl -q -w vm.drop_caches=3'
as root on fex.
[ 28.090052] NX-OS starts punching watchdog
fex login:

```

The following is an example output for configuring secure erase factory reset command on module:

```

switch# factory-reset [all | module <mod>]
switch# factory-reset [module <3>]
!!!! WARNING !!!!
The factory reset operation will erase ALL persistent storage on the specified module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken to render data non-recoverable. Please, proceed with caution and
understanding that this operation cannot be undone and will leave the system in a
fresh-from-factory state.
!!!! WARNING !!!!
Continue? (y/n) [n] y
A module reload is required for the reset operation to proceed. Please, wait...
...truncated...
Secure erase requested! Please, do not power off module!
>>>> Wiping all storage devices ...
+++ Starting mmc secure erase for /dev/mmcblk0 +++
*** Please, wait - this may take several minutes ***
\
---> SUCCESS
+++ Starting SSD secure erase for /dev/sda +++
*** Please, wait - this may take several minutes ***
\
---> SUCCESS
+++ Starting cmos secure erase +++
\
---> SUCCESS
>>>> Done
+++ Starting nvram secure erase +++
\
---> SUCCESS
>>>> Done

```

The following is an example output logs for configuring secure erase factory reset command on LC:

```

switch# show mod

```

Mod	Ports	Module-Type	Model	Status
1	32	32x40/100G Ethernet Module	N9K-X9732C-FX	ok
22	0	4-slot Fabric Module	N9K-C9504-FM-E	ok

24	0	4-slot Fabric Module	N9K-C9504-FM-E	ok
26	0	4-slot Fabric Module	N9K-C9504-FM-E	ok
27	0	Supervisor Module	N9K-SUP-B+	active *
28	0	Supervisor Module	N9K-SUP-B+	ha-standby
29	0	System Controller	N9K-SC-	active
30	0	System Controller	N9K-SC-	standby

Mod	Sw	Hw	Slot
1	10.2(1.196)	0.1070	LC1
22	10.2(1.196)	1.2	FM2
24	10.2(1.196)	1.2	FM4
26	10.2(1.196)	1.1	FM6
27	10.2(1.196)	1.0	SUP1
28	10.2(1.196)	1.2	SUP2
29	10.2(1.196)	1.4	SC1
30	10.2(1.196)	1.4	SC2

```

switch#
switch# factory-reset mod 1
!!!! WARNING !!!!
The factory reset operation will erase ALL persistent storage on the specified module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable.
Please, proceed with
caution and understanding that this operation cannot be undone and will leave the system
in
a fresh-from-factory state.
!!!! WARNING !!!!
Continue? (y/n) [n] y
A module reload is required for the reset operation to proceed. Please, wait...
reloading module 1 ...
.....
SUCCESS! All persistent storage devices detected on the specified module have been purged.

```

```

switch#
switch# show mod

```

Mod	Ports	Module-Type	Model	Status
1	32	32x40/100G Ethernet Module	N9K-X9732C-FX	powered-dn
22	0	4-slot Fabric Module	N9K-C9504-FM-E	ok
24	0	4-slot Fabric Module	N9K-C9504-FM-E	ok
26	0	4-slot Fabric Module	N9K-C9504-FM-E	ok
27	0	Supervisor Module	N9K-SUP-B+	active *
28	0	Supervisor Module	N9K-SUP-B+	ha-standby
29	0	System Controller	N9K-SC-A	active
30	0	System Controller	N9K-SC-A	standby

Mod	Power-Status	Reason
1	powered-dn	Configured Power down

Mod	Sw	Hw	Slot
22	10.2(1.196)	1.2	FM2
24	10.2(1.196)	1.2	FM4
26	10.2(1.196)	1.1	FM6
27	10.2(1.196)	1.0	SUP1
28	10.2(1.196)	1.2	SUP2
29	10.2(1.196)	1.4	SC1

```
switch#
```

The following is an example output logs for configuring secure erase factory reset command on mod:

```

switch# factory-reset mod 26
!!!! WARNING !!!!

```

```
switch# show mod
```

Mod	Power-Status	Reason
1	powered-dn	Configured Power down
26	powered-dn	Configured Power down

```
switch#
```



APPENDIX A

IETF RFCs supported by Cisco NX-OS System Management

This appendix lists the IETF RFCs for system management supported in Cisco NX-OS.

- [IETF RFCs Supported by Cisco NX-OS System Management, on page 581](#)

IETF RFCs Supported by Cisco NX-OS System Management

This appendix lists the IETF RFCs for system management supported in Cisco NX-OS.

RFCs	Title
RFC 2819	<i>Remote Network Monitoring Management Information Base</i>
RFC 3411 and RFC 3418	<i>An Architecture for Describing Simple Network Management (SNMP) Management Frameworks</i>



APPENDIX **B**

Embedded Event Manager System Events and Configuration Examples

This appendix describes the Embedded Event Manager (EEM) system policies, events, and policy configuration examples.

This appendix includes the following sections:

- [EEM System Policies, on page 583](#)
- [EEM Events, on page 586](#)
- [Configuration Examples for EEM Policies, on page 587](#)

EEM System Policies

The following table lists the Embedded Event Manager (EEM) system policies.

Event	Description
__BootupPortLoopback	Do CallHome, Error-disable affected ports, log error testing on affected ports after 1 consecutive failures of GOLD "BootupPortLoopback" test
__PortLoopback	Do CallHome, log error in Syslog/OBFL/Exception Log, and disable further HM testing on affected ports after 10 consecutive failures of GOLD "PortLoopback" test
__RewriteEngineLoopback	Do CallHome, log error in Syslog/OBFL/Exception Log, and disable further HM testing on affected ports after 10 consecutive failures of GOLD "RewriteEngine" test

Event	Description
__asicmem	Do CallHome and log error when GOLD "AsicMemory" test fails. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. Note To avoid a kernel panic when the test fails, you can override the EEM system policy.
__asic_register_check	Do CallHome, log error, and disable further HM testing for that ASIC device/instance after 20 consecutive failures of GOLD "ASICRegisterCheck" test
__compact_flash	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "CompactFlash" test
__crypto_device	Do CallHome and log error when GOLD "CryptoDevice" test fails
__eobc_port_loopback	Do CallHome and log error when GOLD "EOBCPortLoopback" test fails
__ethpm_debug_1	Action: none
__ethpm_debug_2	Action: none
__ethpm_debug_3	Action: none
__ethpm_debug_4	Action: none
__ethpm_link_flap	More than 30 link flaps in a 420-second interval. Action: Error. Disable the port
__external_compact_flash	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "ExternalCompactFlash" test
__fpgareg	Do CallHome, log error, disable further HM testing after 20 consecutive failures of GOLD "FpgaRegTest" test. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. Note To avoid a kernel panic when the test fails, you can override the EEM system policy.

Event	Description
__L2ACLRedirect	Do CallHome, log error, disable further HM testing after 10 consecutive failures of L2ACLRedirect test. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. Note To avoid a kernel panic when the test fails, you can override the EEM system policy.
__lcm_module_failure	Power cycle two times and then power down
__management_port_loopback	Do CallHome and log error when GOLD "ManagementPortLoopback" test fails
__nvram	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "NVRAM" test
__pfm_fanabsent_all_systemfan	Shuts down if both fan trays (f1 and f2) are absent for 2 minutes
__pfm_fanbad_all_systemfan	Syslog when fan goes bad
__pfm_fanbad_any_singlefan	Syslog when fan goes bad
__pfm_power_over_budget	Syslog warning for insufficient power overbudget
__pfm_tempev_major	TempSensor Major Threshold. Action: Shutdown
__pfm_tempev_minor	TempSensor Minor Threshold. Action: Syslog
__primary_bootrom	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "PrimaryBootROM" test
__pwr_mgmt_bus	Do CallHome, log error, and disable further HM testing for the module or spine-card after 20 consecutive failures of GOLD "PwrMgmtBus" test
__real_time_clock	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "RealTimeClock" test
__secondary_bootrom	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "SecondaryBootROM" test
__spine_control_bus	Do CallHome, log error, and disable further HM testing for that module or spine-card after 20 consecutive failures of GOLD "SpineControlBus" test

Event	Description
__standby_fabric_loopback	Do CallHome, log error, and disable further HM testing after 10 consecutive failures
__status_bus	Do CallHome, log error, and disable further HM testing after 5 consecutive failures of GOLD "StatusBus" test
__system_mgmt_bus	Do Call Home, log error, and disable further HM testing for that fan or power supply after 20 consecutive failures of GOLD "SystemMgmtBus" test
__usb	Do Call Home and log error when GOLD "USB" test fails

EEM Events

The following table describes the EEM events you can use on the device.

EEM Event	Description
application	Publishes an application-specific event.
cli	CLI command is entered that matches a pattern with a wildcard.
counter	EEM counter reaches a specified value or range.
fanabsent	System fan tray is absent.
fanbad	System fan generates a fault.
fib	Monitors routes or TCAM usage in the unicast FIB.
gold	GOLD test failure condition is hit.
interface	Interface counter exceeds a threshold.
memory	Available system memory exceeds a threshold.
module	Specified module enters the selected status.
module-failure	Module failure is generated.
none	Runs the policy event without any events specified.
oir	Online insertion or removal occurs.
policy-default	Default parameters and thresholds are used for the events in the system policy you override.

EEM Event	Description
poweroverbudget	Platform software detects a power budget condition.
snmp	SNMP object ID (OID) state changes.
storm-control	Platform software detects an Ethernet packet storm condition.
syslog	Monitors syslog messages and invokes the policy based on the search string in the policy.
sysmgr	System manager generates an event.
temperature	Temperature level in the system exceeds a threshold.
timer	Specified time is reached.
track	Tracked object changes state.

Configuration Examples for EEM Policies

Configuration Examples for CLI Events

Monitoring Interface Shutdown

This example shows how to monitor an interface shutdown:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# event manager applet monitorShutdown
switch(config-applet)#
switch(config-applet)# description "Monitors interface shutdown."
switch(config-applet)# event cli match "conf t; interface *; shutdown"
switch(config-applet)# action 1.0 cli show interface e 3/1
switch(config)# copy running-config startup-config
```



Note Outputs of **show** commands entered as part of EEM policy are archived in the logflash as text files with the "eem_archive_" prefix. To view the archived output, use the **show file logflash:eem_archive_n** command.

Monitoring Module Powerdown

This example shows how to monitor a module powerdown:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# event manager applet monitorPoweroff
switch(config-applet)#
switch(config-applet)# description "Monitors module power down."
switch(config-applet)# event cli match "conf t; poweroff *"
```

```
switch(config-applet)# action 1.0 cli show module
switch(config)# copy running-config startup-config
```

Adding a Trigger to Initiate a Rollback

This example shows how to add a trigger to initiate a rollback:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)#
switch(config)# event manager applet rollbackTrigger
switch(config-applet)#
switch(config-applet)# description "Rollback trigger."
switch(config-applet)# event cli match "rollback *"
switch(config-applet)# action 1.0 cli copy running-config bootflash:last_config
switch(config)# copy running-config startup-config
```

Configuration Examples to Override (Disable) Major Thresholds

Preventing a Shutdown When Reaching a Major Threshold

This example shows how to prevent a shutdown caused by reaching a major threshold:

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Disabling One Bad Sensor

This example shows how to disable only sensor 3 on module 2 when sensor 3 is malfunctioning (all other sensors are unaffected):

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Disabling Multiple Bad Sensors

This example shows how to disable sensors 5, 6, and 7 on module 2 when these sensors are malfunctioning (all other sensors are unaffected):

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 5 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 6 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Overriding (Disabling) an Entire Module

This example shows how to disable module 2 when it is malfunctioning:

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 threshold major
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Overriding (Disabling) Multiple Modules and Sensors

This example shows how to disable sensors 3, 4, and 7 on module 2 and all sensors on module 3 when they are malfunctioning:

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 4 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 3 threshold major
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

Enabling One Sensor While Disabling All Remaining Sensors of All Modules

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Enabling One Sensor While Disabling All Remaining Sensors of All Modules

This example shows how to disable all sensors on all modules except sensor 4 on module 9:

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 4 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

Enabling Multiple Sensors While Disabling All Remaining Sensors of All Modules

This example shows how to disable all sensors on all modules except sensors 4, 6, and 7 on module 9:

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 4 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 6 threshold major
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 7 threshold major
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
```

Enabling All Sensors of One Module While Disabling All Sensors of the Remaining Modules

This example shows how to disable all sensors on all modules except all sensors on module 9:

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

Enabling a Combination of Sensors on Modules While Disabling All Sensors of the Remaining Modules

This example shows how to disable all sensors on all modules except sensors 3, 4, and 7 on module 2 and all sensors on module 3:

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 4 threshold major
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet5 override __pfm_tempev_major
switch(config-applet)# event temperature module 3 threshold major
switch(config-applet)# action 5 policy-default
switch(config-applet)# end
```

Configuration Examples to Override (Disable) Shutdown for Fan Tray Removal

Overriding (Disabling) a Shutdown for Removal of One or More Fan Trays

This example shows how to disable a shutdown so that you can remove one or more (or all) fan trays:

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
```

Overriding (Disabling) a Shutdown for Removal of a Specified Fan Tray

This example shows how to disable a shutdown so that you can remove a specified fan tray (fan tray 3):

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

Overriding (Disabling) a Shutdown for Removal of Multiple Specified Fan Trays

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config)# end
```

Overriding (Disabling) a Shutdown for Removal of Multiple Specified Fan Trays

This example shows how to disable a shutdown so that you can remove multiple specified fan trays (fan trays 2, 3, and 4):

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 4 time 60
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config)# end
```

Overriding (Disabling) a Shutdown for Removal of All Fan Trays Except One

This example shows how to disable a shutdown so that you can remove all fan trays except one (fan tray 2):

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

Overriding (Disabling) a Shutdown for Removal of Fan Trays Except for a Specified Set of Fan Trays

This example shows how to disable a shutdown so that you can remove fans except for a specified set of fan trays (fan trays 2, 3, and 4):

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2,3,4 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

Overriding (Disabling) a Shutdown for Removal of All Fan Trays Except One from a Set of Fan Trays

This example shows how to disable a shutdown so that you can remove all fan trays except one from a set of fan trays (fan trays 2, 3, or 4):

```

switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 4 time 60
switch(config-applet)# action 4 policy-default
switch(config-applet)# end

```

Configuration Examples to Create a Supplemental Policy

Creating a Supplemental Policy for the Fan Tray Absent Event

This example shows how to create a supplemental policy using the **event fanabsent** command:

```
[no] event fanabsent [fan fan-tray-number] time time-interval
```

In addition to the default policy, this example shows how to execute the policy myappletname and action 3 if fan tray 1 is absent for 60 seconds:

```

switch# configure terminal
switch(config)# event manager applet myappletname
switch(config-applet)# event fanabsent fan 1 time 60
switch(config-applet)# action 3 cli "show env fan"
switch(config-applet)# end

```

Creating a Supplemental Policy for the Temperature Threshold Event

This example shows how to create a supplemental policy using the **event temperature** command:

```
[no] event temperature [mod module-number] [sensor sensor-number] threshold {major | minor | any}
```

In addition to the default policy, this example shows how to execute the policy myappletname and action 1 if the temperature crosses the minor threshold on sensor 3 of module 2:

```

switch# configure terminal
switch(config)# event manager applet myappletname
switch(config-applet)# event temperature module 2 sensor 3 threshold minor
switch(config-applet)# action 1 cli "show environ temperature"
switch(config-applet)# end

```

Configuration Examples for the Power Over-Budget Policy

The power over-budget policy gets triggered when the available power capacity drops below zero and the device is no longer able to keep the previously powered-up modules in the powered-up state. The default action is to print a syslog to notify the user of the occurrence of power over budget.

You can enable an additional action to power down modules until the available power recovers from the red (negative) zone.

Shutting Down Modules

If you do not specify any modules, the power over-budget shutdown starts from slot 1 and shuts down modules until the power recovers from the red (negative) zone. Empty slots and slots that contain a supervisor, standby supervisor, spine, or crossbar are skipped.

This example shows how to shut down modules starting from module 1 when the available power drops below zero:

```
switch# configure terminal
switch(config)# event manager applet <myappletname4a> override __pfm_power_over_budget
switch(config-applet)# event poweroverbudget
switch(config-applet)# action 4 overbudgetshut
switch(config-applet)# end
```

Shutting Down a Specified List of Modules

You can specify a list of modules that the power over-budget action uses to shut down modules until the power recovers from the red (negative) zone. Empty slots and slots that contain a supervisor, standby supervisor, spine, or crossbar are skipped.

This example shows how to shut down modules from a specified list of modules (1, 2, 7, 8) when the available power drops below zero:

```
switch# configure terminal
switch(config)# event manager applet <myappletname4b> override __pfm_power_over_budget
switch(config-applet)# event poweroverbudget
switch(config-applet)# action 5 overbudgetshut module 1,2,7,8
switch(config-applet)# end
```

Configuration Examples to Select Modules to Shut Down

Using the Policy Default to Select Nonoverridden Modules to Shut Down

This example shows how to use the policy default to select the nonoverridden modules to shut down when a major threshold is exceeded:

```
switch# configure terminal
switch(config)# event manager applet my5a1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet my5a2 override __pfm_tempev_major
switch(config-applet)# event temperature module 1-3 sensor 4 threshold major
switch(config-applet)# action 5 policy-default
switch(config-applet)# end
```

Using Parameter Substitution to Select Nonoverridden Modules to Shut Down

This example shows how to use parameter substitution to select the nonoverridden modules to shut down when a major threshold is exceeded:

```
switch# configure terminal
switch(config)# event manager applet my5b1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
```



```
switch(config)# event manager applet my5b2 override __pfm_tempev_major
switch(config-applet)# event temperature module 1-3 sensor 8 threshold major
switch(config-applet)# action 6 forceshut module my_module_list reset "temperature-sensor
policy trigger"
switch(config-applet)# end
```

To create event manager parameters, use the **event manager environment** command. To display the values of event manager parameters, use the **show event manager environment all** command.

Configuration Examples for the Online Insertion Removal Event

The online insertion removal (OIR) event does not have a default policy.

This example shows how to configure the OIR event using the **event oir** command:

event oir *device-type event-type [device-number]*

The *device-type* can be **fan**, **module**, or **powersupply**.

The *event-type* can be **insert**, **remove**, or **anyoir** (insert or remove).

The optional *device-number* specifies a single device. If omitted, all devices are selected.

This example shows how to configure the insert event:

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module insert
switch(config-applet)# action 1 syslog priority critical msg "OIR insert event: A Module
is inserted"
```

This example shows how to configure the remove event:

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module remove
switch(config-applet)# action 1 syslog priority critical msg "OIR remove event: A Module
is removed"
```

Configuration Example to Generate a User Syslog

This example shows how to generate a user syslog using the **action syslog** command:

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module remove
switch(config-applet)# action 1 syslog priority critical msg "Module is removed"
```

When this event is triggered, the system generates a syslog as follows:

```
switch(config)# 2013 May 20 00:08:27 plb-57 %$ VDC-1 %$ %EEM_ACTION-2-CRIT: "Module is
removed"
```

Configuration Example to Monitor Syslog Messages

This example shows how to monitor syslog messages from the switch:

```
switch(config)# event manager applet a1
switch(config-applet)# event syslog occurs 6 period 4294967 pattern "authentication failed"
```

When this event is triggered, the action defined in the policy is executed.

Configuration Examples for SNMP Notification

Polling an SNMP OID to Generate an EEM Event

The SNMP object ID (OID) CISCO-SYSTEM-EXT-MIB::cseSysCPUUtilization is used for querying the CPU utilization of the switch:

```
cseSysCPUUtilization OBJECT-TYPE
SYNTAX Gauge32 (0..100 )
UNITS "%"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The average utilization of CPU on the active supervisor."
::= { ciscoSysInfoGroup 1 }
```

This example shows the use of an SNMP OID that is polled at an interval of 10 seconds and has a threshold value of 95 percent:

```
switch# configure terminal
switch(config)# event manager applet test_policy
switch(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.305.1.1.1.0 get-type exact entry-op
gt entry-val 95 exit-op lt exit-val 90 poll-interval 10
```

Sending an SNMP Notification in Response to an Event in the Event Policy

You can use this type of configuration to cause a critical event trigger to generate an SNMP notification.

This example shows how to send an SNMP notification for an event from the Event Manager applet configuration mode:

```
switch(config-applet)# action 1.1 snmp-trap intdata1 100 intdata2 300 strdata "CPU Hogging
at switch1"
switch(config-applet)# action 1.1 snmp-trap intdata1 100 intdata2 300 strdata "Port Failure
eth9/1"
```

This configuration triggers an SNMP notification (trap) from the switch to SNMP hosts. The SNMP payload carries the values of user-defined fields intdata1, intdata2, and strdata.

Configuration Example for Port Tracking

This example shows how to configure the state of one port to match the state of another port (port tracking).

To configure the port tracking of Ethernet interface 3/23 by Ethernet interface 1/2, follow these steps:

Procedure

-
- Step 1** Create an object to track the status of Ethernet interface 3/23.

Example:

```
switch# configure terminal
switch(config)# track 1 interface ethernet 3/23
switch(config-track)# end
```

Step 2 Configure an EEM event to shut Ethernet interface 1/2 when the tracking object shuts down.

Example:

```
switch(config)# event manager applet track_3_23_down
switch(config-applet)# event track 1 state down
switch(config-applet)# action 1 syslog msg EEM applet track_3_23_down shutting down port
eth1/2 due to eth3/23 being down
switch(config-applet)# action 2 cli conf term
switch(config-applet)# action 3 cli interface ethernet 1/2
switch(config-applet)# action 4 cli shut
switch(config-applet)# end
```

Step 3 Configure an EEM event to bring up Ethernet interface 1/2 when Ethernet interface 3/23 comes up.

Example:

```
switch# configure terminal
switch(config)# event manager applet track_3_23_up
switch(config-applet)# event track 1 state up
switch(config-applet)# action 1 syslog msg EEM applet track_3_23_down bringing up port
eth1/2 due to eth3/23 being up
switch(config-applet)# action 2 cli conf term
switch(config-applet)# action 3 cli interface ethernet 1/2
switch(config-applet)# action 4 cli no shut
switch(config-applet)# end
```

Configuration Example to Register an EEM Policy with the EEM

This example shows how to register an EEM policy with the EEM:

Basic switch configuration:

```
event manager applet vpc_check_peer_at_startup
event track 101 state up
action 1.0 cli copy bootflash:eem/user_script_policies/load_schedules running-config

feature scheduler

!!## 2 x dummy loopbacks are required ###
interface loopback 101
interface loopback 102

track 1 list boolean or
object 13
object 12
object 102
track 2 list boolean and
object 13
object 12
track 12 interface Ethernet 2/24 line-protocol
track 13 interface port-channel 3000 line-protocol
```

```
track 101 interface loopback 101 line-protocol
track 102 interface loopback 102 line-protocol
```



Note In this example, port channel 3000 is the vPC peer link, and Ethernet 2/24 is the vPC keepalive link.

You need to copy the following files to the bootflash:

- A directory called: /eem/user_script_policies needs to be created on the supervisor bootflash.
- These five files need to be created and loaded into the above directory:
 - load_schedules
 - remove_vpc_if_peer_failed
 - clean_up
 - unload_schedules
 - restore_vpc

Configuration for the load_schedules file:

```
feature scheduler

configure terminal
scheduler job name vpc_check
configure terminal
event manager policy remove_vpc_if_peer_failed
end

configure terminal
scheduler job name clean_up
configure terminal
event manager policy clean_up
end

configure terminal
scheduler job name trigger
configure terminal
interface loopback 102
shutdown
no shutdown
end

configure terminal
scheduler schedule name load_vpc_check
time start +00:00:04
job name vpc_check

scheduler schedule name trigger_vpc_check
time start +00:00:05
job name trigger

scheduler schedule name load_clean_up
time start +00:00:08
job name clean_up
```

```
scheduler schedule name trigger_clean_up
time start +00:00:10
job name trigger
```

Configuration for the remove_vpc_if_peer_failed file:

```
event manager applet remove_vpc_if_peer_failed
event track 1 state down
action 1.0 cli show run vpc > bootflash://sup-active/eem/user_script_policies/vpc_saved.cfg
action 2.0 cli show run vpc > bootflash://sup-standby/eem/user_script_policies/vpc_saved.cfg
action 3.0 cli configure terminal
action 4.0 cli no feature vpc
action 5.0 syslog msg severity alert "##### WARNING!!!! PEER SWITCH FAILED TO COME ONLINE.
VPC CONFIG REMOVED #####"
action 6.0 cli event manager policy restore_vpc
action 7.0 cli copy bootflash:eem/user_script_policies/unload_schedules running-config
action 8.0 cli no event manager applet remove_vpc_if_peer_failed
action 9.0 cli end
```

Configuration for the clean_up file:

```
event manager applet clean_up
event track 102 state up
action 1.0 cli configure terminal
action 2.0 cli no event manager applet remove_vpc_if_peer_failed
action 3.0 cli copy bootflash:eem/user_script_policies/unload_schedules running
action 4.0 cli no event manager applet clean_up
action 5.0 end
```

Configuration for the unload_schedules file:

```
no scheduler schedule name load_vpc_check
no scheduler schedule name trigger_vpc_check
no scheduler schedule name load_clean_up
no scheduler schedule name trigger_clean_up
no scheduler job name vpc_check
no scheduler job name trigger
no scheduler job name clean_up
```

Configuration for the restore_vpc file:

```
event manager applet restore_vpc
event track 2 state up
action 1.0 cli copy bootflash:eem/user_script_policies/vpc_saved.cfg running-config
action 1.0 syslog priority alerts msg VPC PEER DETECTED. VPC CONFIG RESTORED
action 3.0 cli configure terminal
action 4.0 cli copy bootflash:eem/user_script_policies/unload_schedules running-config
action 5.0 cli no event manager applet restore_vpc
action 6.0 cli end
```



Note The severity keyword is deprecated and only the following patterns are allowed:

```
[0-9 a-zA-Z][0-9 a-zA-Z]*[-_ ,/0-9a-zA-Z-Z]*
```




APPENDIX

Configuration Limits for Cisco NX-OS System Management

The configuration limits are documented in the *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.

- [Configuration Limits for Cisco NX-OS System Management, on page 601](#)

Configuration Limits for Cisco NX-OS System Management

The features supported by Cisco NX-OS have maximum configuration limits. Some of the features have configurations that support limits less than the maximum limits.

The configuration limits are documented in the *Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*.



INDEX

A

abort [32, 217](#)
 action [302, 307, 309, 311](#)
 alert-group {Configuration | Diagnostic | EEM | Environmental |
 Inventory | License | Supervisor-Hardware | Syslog-group-port
 | System | Test} user-def-cmd [188](#)

C

callhome [182–183, 185–186, 188–195, 197](#)
 callhome send [198](#)
 callhome send configuration [198](#)
 callhome send diagnostic [198](#)
 callhome test [198](#)
 cdp advertise {v1 | v2} [151](#)
 cdp enable [150](#)
 cdp format device-id {mac-address | serial-number | system-name} [151](#)
 cdp holdtime [151](#)
 cdp timer [151](#)
 cfs ipv4 distribute [27](#)
 checkpoint [564](#)
 clear cdp counters [152](#)
 clear cdp table [152](#)
 clear checkpoint database [565](#)
 clear counters interface all [465](#)
 clear counters mpls strip [482](#)
 clear hardware rate-limiter sflow [465](#)
 clear lldp counters [423](#)
 clear logging logfile [170](#)
 clear logging nvram [171](#)
 clear logging onboard [350](#)
 clear monitor session all stats [377, 404](#)
 clear monitor session stats [377, 404](#)
 clear mpls strip label dynamic [482](#)
 clear ntp session [144](#)
 clear ntp statistics [144](#)
 clear scheduler logfile [227](#)
 clear sflow statistics [465](#)
 Clock protocol gnss [80](#)
 collect [434](#)
 collect counter [435](#)
 collect ip version [435](#)
 collect timestamp sys-uptime [435](#)
 collect transport tcp flags [436](#)

commit [28, 30, 32, 183–184, 186–194, 216](#)
 config sync [27, 29, 31, 33](#)
 configuration example [407](#)
 ERSPAN [407](#)
 destination [407](#)
 destination over ipv6 [407](#)
 configure maintenance profile maintenance-mode [509](#)
 configure maintenance profile normal-mode [511](#)
 configure session [215](#)
 contract-id [182](#)
 copy ftp [530](#)
 copy sftp [531](#)
 copy tftp [529](#)
 customer-id [182](#)

D

description [302, 311, 367, 391, 433, 436, 438](#)
 destination [436](#)
 destination interface [368, 373](#)
 destination ip [392](#)
 destination-profile [184–185, 187, 195](#)
 destination-profile {CiscoTAC-1 | full-txt-destination |
 short-txt-destination} alert-group [187](#)
 destination-profile {CiscoTAC-1 | full-txt-destination |
 short-txt-destination} email-addr [185](#)
 destination-profile {CiscoTAC-1 | full-txt-destination |
 short-txt-destination} http [185](#)
 destination-profile {CiscoTAC-1 | full-txt-destination |
 short-txt-destination} message-level [185](#)
 destination-profile {CiscoTAC-1 | full-txt-destination |
 short-txt-destination} message-size [186](#)
 destination-profile {CiscoTAC-1 | full-txt-destination |
 short-txt-destination} transport-method {email | http} [185](#)
 destination-profile name alert-group all [195](#)
 destination-profile name email-address email-address [195](#)
 diagnostic bootup level {complete | minimal | bypass} [290](#)
 diagnostic clear result module [293](#)
 diagnostic monitor interval module [291](#)
 diagnostic monitor module [291](#)
 diagnostic ondemand action-on-failure {continue failure-count |
 stop} [292](#)
 diagnostic ondemand iteration [292](#)
 diagnostic start module [292](#)
 diagnostic stop module [292](#)

diagnostic test simulation module [292–293](#)
 dir [528](#)
 dir bootflash: [532](#)
 dscp [436](#)

E

email-contact [182, 195](#)
 enable [194, 197](#)
 ERSPAN [401, 407](#)
 configuring destination sessions [401](#)
 destination [407](#)
 configuration example [407](#)
 destination over ipv6 [407](#)
 configuration example [407](#)
 destination sessions [401](#)
 configuring for ERSPAN [401](#)
 erspan-id [392](#)
 event [302, 311](#)
 event application [303](#)
 event cli [303](#)
 event counter [303](#)
 event fanabsent [303](#)
 event fanbad [303](#)
 event fib adjacency extra [304](#)
 event fib resource team usage [304](#)
 event fib route {extra | inconsistent | missing} [304](#)
 event gold module [304](#)
 event manager applet [302, 310, 313](#)
 event manager environment [301](#)
 event manager policy [310](#)
 event memory {critical | minor | severe} [304](#)
 event module [304](#)
 event module-failure [304](#)
 event none [304](#)
 event oir [305](#)
 event policy-default count [305](#)
 event poweroverbudget [305](#)
 event snmp [305](#)
 event storm-control [305](#)
 event syslog [306](#)
 event syslog {occurs | period | pattern | priority} [313](#)
 event syslog tag {occurs | period | pattern | priority} [313](#)
 event sysmgr memory [306](#)
 event sysmgr switchover count [306](#)
 event temperature [306](#)
 event timer [307](#)
 event track [307](#)
 exporter [438](#)

F

feature lldp [413](#)
 feature netflow [433](#)
 feature ntp [135](#)

feature ptp [75](#)
 feature scheduler [221](#)
 feature sflow [457](#)
 filter access-group [368](#)
 flow exporter [436](#)
 flow monitor [438](#)
 flow record [433, 440–441](#)

G

guestshell [545](#)
 guestshell sync [545](#)

H

hardware acl tap-agg [472](#)
 hardware multicast global-tx-span [373](#)
 hw-module logging onboard [347](#)
 hw-module logging onboard counter-stats [347](#)
 hw-module logging onboard cpuhog [347](#)
 hw-module logging onboard environmental-history [347](#)
 hw-module logging onboard error-stats [348](#)
 hw-module logging onboard interrupt-stats [348](#)
 hw-module logging onboard module [348](#)
 hw-module logging onboard obfl-logs [348](#)

I

import [31](#)
 import interface [31](#)
 import running-config [31](#)
 install activate [533](#)
 install add bootflash [532](#)
 install add ftp [532](#)
 install add tftp [532](#)
 install add usb1 [532](#)
 install add usb2 [532](#)
 install commit [533–534](#)
 install deactivate [534](#)
 install remove [535](#)
 ip access-group [216](#)
 ip access-list [215, 371, 399, 473](#)
 ip dscp [392](#)
 ip flow monitor [439, 442](#)
 ip port access-group [474, 476, 479](#)
 ip ttl [392](#)
 ipv6 flow monitor [439, 442](#)
 isolate [506](#)

J

job name [226](#)

L

layer2-switched flow monitor [441](#)
 lldp chassis-id switch [413](#)
 lldp dcbox version [416](#)
 lldp holdtime [421](#)
 lldp receive [414, 420](#)
 lldp reinit [421](#)
 lldp timer [421](#)
 lldp tlv-select [422](#)
 lldp transmit [414, 420](#)
 logging console [157](#)
 logging event {link-status | trunk-status} {enable | default} [161](#)
 logging level [162–163](#)
 logging logfile [159](#)
 logging message interface type ethernet description [158](#)
 logging module [161](#)
 logging monitor [157](#)
 logging origin-id [158](#)
 logging server [165–166](#)
 logging source-interface loopback [166](#)
 logging timestamp {microseconds | milliseconds | seconds} [163](#)

M

mac access-list [473](#)
 mac packet-classify [440](#)
 mac port access-group [474, 476, 479](#)
 match [433](#)
 match datalink [434, 440](#)
 match ip [434](#)
 match ipv4 [435](#)
 match ipv6 [435](#)
 match transport [435](#)
 mode tap-aggregation [476](#)
 monitor erspan origin ip-address [390](#)
 monitor session [367, 377, 390, 393](#)
 monitor session all shut [377, 393](#)
 monitor session all type erspan-source [390](#)
 monitor session in erspan source [400](#)
 mpls strip [478](#)
 mpls strip dest-mac [481](#)
 mpls strip label [480](#)
 mpls strip label-age [481](#)
 mtu [372](#)

N

NetFlow [439, 442](#)
 bridged on VLAN [439](#)
 timeouts [442](#)
 no duplicate-message throttle [193](#)
 no isolate [506](#)
 no monitor session [367, 390, 394](#)
 no monitor session all shut [394](#)

no scheduler job name [225](#)
 no shut [369, 392, 394](#)
 no shutdown [506](#)
 no snmp trap link-status [264](#)
 no snmp-server protocol enable [268](#)
 no switch-profile [33](#)
 no system interface shutdown [506](#)
 no system mode maintenance [518](#)
 no system mode maintenance dont-generate-profile [518](#)
 no system mode maintenance on-reload reset-reason [515](#)
 ntp access-group {peer | serve | serve-only | query-only} [141](#)
 ntp authenticate [140](#)
 ntp authentication-key [139](#)
 ntp logging [143](#)
 ntp master [136](#)
 ntp peer [137](#)
 ntp server [137](#)
 ntp source [142](#)
 ntp source-interface [143](#)
 ntp trusted-key [139](#)

O

option exporter-stats [437](#)
 option interface-table [437](#)

P

periodic-inventory notification [192](#)
 periodic-inventory notification interval [192](#)
 periodic-inventory notification timeofday [192](#)
 permit [215, 473](#)
 permit ip [371, 399](#)
 permit udf [371, 399](#)
 phone-contact [182](#)
 ptp [81, 88](#)
 ptp announce {interval | timeout} [84](#)
 ptp clock-mode [77](#)
 ptp delay-request minimum interval [84–85](#)
 ptp device-type boundary-clock [76, 80](#)
 ptp device-type generalized-ptp [76, 80](#)
 ptp device-type ordinary-clock-grandmaster [76, 80](#)
 ptp domain [76–77](#)
 ptp priority1 [77](#)
 ptp priority2 [77](#)
 ptp sync interval [85](#)
 Ptp utc-offset [80](#)
 ptp vlan [85](#)
 python instance [507, 510](#)

R

record [438](#)
 reload [371, 374, 399](#)
 rmon alarm [278](#)

rmon event [279](#)
 rmon hcalarm [278](#)
 rollback running-config {checkpoint | file} [565](#)
 run bash [543](#)

S

save [215–216](#)
 scheduler aaa-authentication password [223](#)
 scheduler aaa-authentication username [223](#)
 scheduler job name [224](#)
 scheduler logfile size [222](#)
 scheduler schedule name [226](#)
 sflow agent-ip [462](#)
 sflow collector-ip [461](#)
 sflow collector-port [462](#)
 sflow counter-poll-interval [459](#)
 sflow data-source interface ethernet [463](#)
 sflow data-source interface port-channel [463](#)
 sflow max-datagram-size [460](#)
 sflow max-sampled-size [458](#)
 sflow sampling-rate [458](#)
 show callhome [183, 190, 192, 198](#)
 show callhome destination-profile [184, 186–187, 198](#)
 show callhome destination-profile profile [184, 186–187](#)
 show callhome transport [189, 191, 198](#)
 show callhome user-def-cmds [188, 198](#)
 show cdp all [152](#)
 show cdp entry {all | name} [152](#)
 show cdp global [152](#)
 show cdp interface [150, 152](#)
 show cdp neighbors {device-id | interface} [152](#)
 show cdp neighbors detail [148](#)
 show checkpoint [564–565](#)
 show checkpoint all [565](#)
 show checkpoint all system [565](#)
 show checkpoint all user [565](#)
 show checkpoint summary [565](#)
 show checkpoint summary system [565](#)
 show checkpoint summary user [565](#)
 show clock [527](#)
 show configuration session [215–217](#)
 show configuration session status [217](#)
 show configuration session summary [217](#)
 show diagnostic bootup level [290, 293](#)
 show diagnostic content module [291, 293](#)
 show diagnostic description module [293](#)
 show diagnostic events [293](#)
 show diagnostic ondemand setting [293](#)
 show diagnostic result module [293](#)
 show diagnostic simulation module [293](#)
 show diagnostic status module [292–293](#)
 show diff rollback-patch {checkpoint | running-config | startup-config | file} [565](#)
 show event manager environment [301, 314](#)
 show event manager environment all [301, 314](#)
 show event manager event-types [314](#)
 show event manager event-types all [314](#)
 show event manager event-types module [314](#)
 show event manager history events [314](#)
 show event manager policy-state [302, 311, 314](#)
 show event manager script system [314](#)
 show event manager script system all [314](#)
 show event manager system-policy [296, 301, 314](#)
 show event manager system-policy all [314](#)
 show feature [457](#)
 show flow cache [443, 449](#)
 show flow event [450](#)
 show flow exporter [443, 450](#)
 show flow filter [450](#)
 show flow interface [443, 450](#)
 show flow monitor [450](#)
 show flow profile [450](#)
 show flow record [434, 443, 450](#)
 show flow record netflow layer2-switched input [441, 443](#)
 show flow system [450](#)
 show hardware capacity [293](#)
 show install active [527, 533](#)
 show install committed [533](#)
 show install inactive [532, 534](#)
 show install log [533, 544](#)
 show install log detail [544](#)
 show install packages [543](#)
 show interface brief [520](#)
 show interface snmp-ifindex [265, 270](#)
 show ip access-list acl_name [497](#)
 show ip access-lists [474, 476](#)
 show lldp interface [414, 421–422](#)
 show lldp neighbors detail [422](#)
 show lldp neighbors interface [422](#)
 show lldp timers [421–422](#)
 show lldp tlv-select [422](#)
 show lldp traffic [422](#)
 show lldp traffic interface [423](#)
 show logging console [157, 171](#)
 show logging info [161, 171](#)
 show logging last [170–171](#)
 show logging level [162, 171](#)
 show logging logfile [170–171](#)
 show logging logfile end-time [170–171](#)
 show logging logfile start-time [170–171](#)
 show logging module [162, 171](#)
 show logging monitor [158, 171](#)
 show logging nvram [170–171](#)
 show logging nvram last [170–171](#)
 show logging onboard [348](#)
 show logging onboard boot-uptime [349](#)
 show logging onboard counter-stats [349](#)
 show logging onboard credit-loss [349](#)
 show logging onboard device-version [349](#)
 show logging onboard endtime [349](#)
 show logging onboard environmental-history [349](#)

- show logging onboard error-stats 349
- show logging onboard exception-log 349
- show logging onboard interrupt-stats 349
- show logging onboard module 349
- show logging onboard obfl-history 349
- show logging onboard obfl-logs 349
- show logging onboard stack-trace 349
- show logging onboard starttime 349
- show logging onboard status 349
- show logging origin-id 159
- show logging server 166, 171
- show logging timestamp 163, 171
- show mac access-lists 474, 476
- show maintenance on-reload reset-reasons 520
- show maintenance profile 520
- show maintenance profile maintenance-mode 509, 520
- show maintenance profile normal-mode 511, 520
- show maintenance timeout 520
- show module 293, 527
- show monitor 377
- show monitor session 369, 373, 377, 392, 397, 401, 404
- show monitor session all 369, 377, 394, 397, 404
- show monitor session all stats 377, 404
- show monitor session range 369, 377, 397, 404
- show monitor session stats 377, 404
- show ntp access-groups 142, 144
- show ntp authentication-keys 139, 144
- show ntp authentication-status 140, 144
- show ntp logging-status 143–144
- show ntp peer-status 144
- show ntp peers 138, 144
- show ntp rts-update 144
- show ntp source 144
- show ntp source-interface 144
- show ntp statistics {io | local | memory | peer {ipaddr | name}} 144
- show ntp trusted-keys 139, 144
- show process 465
- show ptp brief 86, 113
- show ptp clock 113
- show ptp clock foreign-masters-record 113
- show ptp corrections 113
- show ptp counters 113
- show ptp parent 113
- show ptp port interface 86
- show ptp port interface ethernet 113
- show ptp time-property 113
- show qos dcbxp interface 423
- show rmon {alarms | hcalarms} 278
- show rmon alarms 279
- show rmon events 279
- show rmon hcalarms 279
- show rmon logs 279
- show rollback log 565
- show rollback log exec 565
- show rollback log verify 565
- show run aclmgr 497
- show run ofm 497
- show running-config | include "scheduler aaa-authentication" 223
- show running-config | include "system memory" 312
- show running-config callhome 198
- show running-config eem 296, 314
- show running-config lldp 414, 422
- show running-config mmode 520
- show running-config monitor 393–394, 404
- show running-config netflow 443, 450
- show running-config ntp 136, 144
- show running-config ptp 113
- show running-config sflow 465
- show running-config sflow all 465
- show running-config snmp 270
- show running-config switch-profile 35
- show scheduler config 222, 227
- show scheduler job 224–225, 227
- show scheduler job name 224–225
- show scheduler logfile 227
- show scheduler schedule 227
- show sflow 458–465
- show snapshots 512, 520
- show snapshots compare 512, 520
- show snapshots dump 520
- show snapshots sections 514, 520
- show snmp 266, 270
- show snmp community 271
- show snmp context 267, 271
- show snmp engineID 271
- show snmp group 271
- show snmp host 256, 271
- show snmp session 271
- show snmp source-interface 253, 256, 271
- show snmp trap 271
- show snmp user 247, 271
- show startup-config callhome 198
- show startup-config eem 314
- show startup-config mmode 520
- show startup-config monitor 393–394, 404
- show startup-config switch-profile 35
- show switch-profile 29–30, 32, 34
- show system mode 516, 518, 520
- show tech-support callhome 198
- show tunnel-profile 497
- shut 377, 394
- shutdown 506
- site-id 183
- sleep instance 507
- snapshot create 512
- snapshot delete 512
- snapshot section add 513
- snapshot section delete 514
- snmp-server aaa-user cache-timeout 269
- snmp-server community 250
- snmp-server contact 182, 266
- snmp-server context 267

- snmp-server counter cache timeout [268](#)
- snmp-server enable traps [259](#)
- snmp-server enable traps aaa [259](#)
- snmp-server enable traps bgp [259](#)
- snmp-server enable traps bridge [259](#)
- snmp-server enable traps callhome [260](#)
- snmp-server enable traps config [260](#)
- snmp-server enable traps eigrp [260](#)
- snmp-server enable traps entity [260](#)
- snmp-server enable traps feature-control [261](#)
- snmp-server enable traps hsrp [261](#)
- snmp-server enable traps license [261](#)
- snmp-server enable traps link [262](#)
- snmp-server enable traps ospf [262](#)
- snmp-server enable traps rf [262](#)
- snmp-server enable traps rmon [263](#)
- snmp-server enable traps snmp [263](#)
- snmp-server enable traps stpx [263](#)
- snmp-server enable traps syslog [263](#)
- snmp-server enable traps sysmgr [263](#)
- snmp-server enable traps upgrade [264](#)
- snmp-server enable traps vtp [264](#)
- snmp-server globalEnforcePriv [249](#)
- snmp-server host [251](#), [253–254](#), [256](#)
- snmp-server location [266](#)
- snmp-server mib community-map [267](#)
- snmp-server name [246](#)
- snmp-server source-interface {traps | informs} [253](#)
- snmp-server source-interface traps [255](#)
- snmp-server tcp-session [265](#)
- snmp-server user [249](#), [253](#)
- source [436](#)
- source interface [372](#), [400](#)
- source-interface [197](#)
- SPAN sessions [427](#)
 - configuring [427](#)
- statistics per-entry [473](#)
- storm-control action trap [264](#)
- streetaddress [182](#)
- switch-priority [183](#)
- switch-profile [28–29](#), [31](#)
- switchport [28](#), [366](#), [440–441](#), [474](#), [478](#)
- switchport monitor [366](#)
- sync-peer destination [32](#)

- sync-peers destination [28](#), [33](#)
- system interface shutdown [506](#)
- system memory-thresholds minor [312](#)
- system memory-thresholds threshold critical no-process-kill [312](#)
- system mode maintenance dont-generate-profile [515](#)
- system mode maintenance on-reload reset-reason [515](#)
- system mode maintenance shutdown [515](#)
- system mode maintenance timeout [515](#)

T

- tag [302](#)
- template data timeout [437](#)
- terminal event-manager bypass [309](#)
- terminal monitor [156](#)
- time daily [226](#)
- time monthly [226](#)
- time start [226](#)
- time start now [226](#)
- time start repeat [226](#)
- time weekly [226](#)
- transport email from [189](#)
- transport email from callhome_email-address [195](#)
- transport email mail-server [189](#)
- transport email reply-to [189](#)
- transport email smtp-server hostname/ip-address port 465/587 use-vrf vrf-name [196](#)
- transport email username passwd {cleartext|encrypted} [196](#)
- transport http proxy enable [191](#)
- transport http proxy server [191](#)
- transport http use-vrf [190](#), [196](#)
- transport udp [437](#)

U

- udf [370](#), [398](#)

V

- verify [30](#), [216](#)
- version 9 [437](#)
- vlan configuration [439](#)
- vrf [392](#)