# Cisco Nexus 3000 Series NX-OS System Management Configuration Guide, Release 7.x

**First Published:** 2015-07-22

**Last Modified:** 2024-01-12

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

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

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

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

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

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

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

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: http://www.cisco.com/go/trademarks. 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. (1110R)

© 2017–2024 Cisco Systems, Inc. All rights reserved.

# CONTENTS


PREFACE

**Preface** xvii

Audience xvii

Document Conventions xvii

Related Documentation for Cisco Nexus 3000 Series Switches xviii

Documentation Feedback xviii

Communications, Services, and Additional Information xviii

CHAPTER 1

**New and Changed Information** 1

New and Changed Information 1

CHAPTER 2

**Overview** 3

Licensing Requirements 3

System Management Features 3

CHAPTER 3

**Configuring Switch Profiles** 7

Information About Switch Profiles 7

Switch Profile Configuration Modes 8

Configuration Validation 8

Software Upgrades and Downgrades with Switch Profiles 9

Prerequisites for Switch Profiles 10

Guidelines and Limitations for Switch Profiles 10

Configuring Switch Profiles 11

Adding a Switch to a Switch Profile 13

Adding or Modifying Switch Profile Commands 14

Importing a Switch Profile 17

Verifying Commands in a Switch Profile 19

Isolating a Peer Switch 19
Deleting a Switch Profile 20
Deleting a Switch from a Switch Profile 20
Displaying the Switch Profile Buffer 21
Synchronizing Configurations After a Switch Reboot 22
Switch Profile Configuration show Commands 23
Supported Switch Profile Commands 23
Configuration Examples for Switch Profiles 25
Creating a Switch Profile on a Local and Peer Switch Example 25
Verifying the Synchronization Status Example 26
Displaying the Running Configuration 27
Displaying the Switch Profile Synchronization Between Local and Peer Switches 27
Displaying Verify and Commit on Local and Peer Switches 28
Successful and Unsuccessful Synchronization Examples 29
Configuring the Switch Profile Buffer, Moving the Buffer, and Deleting the Buffer 29

CHAPTER 4 **Using Cisco Fabric Services** 31
Information About CFS 31
CFS Distribution 32
CFS Distribution Modes 32
Uncoordinated Distribution 32
Coordinated Distribution 32
Unrestricted Uncoordinated Distributions 32
Verifying the CFS Distribution Status 33
CFS Support for Applications 33
CFS Application Requirements 33
Enabling CFS for an Application 33
Verifying Application Registration Status 33
Locking the Network 34
Verifying CFS Lock Status 35
Committing Changes 35
Discarding Changes 35
Saving the Configuration 35
Clearing a Locked Session 35

CFS Regions 36
About CFS Regions 36
Example Scenario 36
Managing CFS Regions 36
Creating CFS Regions 36
Assigning Applications to CFS Regions 37
Moving an Application to a Different CFS Region 37
Removing an Application from a Region 38
Deleting CFS Regions 38
Configuring CFS over IP 39
Enabling CFS over IPv4 39
Verifying the CFS Over IP Configuration 39
Configuring IP Multicast Addresses for CFS over IP 39
Configuring IPv4 Multicast Address for CFS 40
Verifying the IP Multicast Address Configuration for CFS over IP 40
Default Settings for CFS 40

CHAPTER 5 **Configuring PTP** 41
Information About PTP 41
PTP Device Types 41
PTP Process 42
High Availability for PTP 43
Guidelines and Limitations for PTP 43
Default Settings for PTP 43
Configuring PTP 44
Configuring PTP Globally 44
Configuring PTP on an Interface 46
Configuring Multiple PTP Domains 47
Configuring clock Identity 50
Configuring PTP Cost Interface 50
Verifying the PTP Configuration 51

CHAPTER 6 **Configuring NTP** 53
Information About NTP 53

NTP as Time Server 54
Distributing NTP Using CFS 54
Clock Manager 54
High Availability 54
Virtualization Support 54
Prerequisites for NTP 55
Guidelines and Limitations for NTP 55
Default Settings 56
Configuring NTP 56
Enabling or Disabling NTP on an Interface 56
Configuring the Device as an Authoritative NTP Server 57
Configuring an NTP Server and Peer 58
Configuring NTP Authentication 59
Configuring NTP Access Restrictions 61
Configuring the NTP Source IP Address 63
Configuring the NTP Source Interface 63
Configuring an NTP Broadcast Server 64
Configuring an NTP Multicast Server 65
Configuring an NTP Multicast Client 66
Configuring NTP Logging 66
Enabling CFS Distribution for NTP 67
Committing NTP Configuration Changes 67
Discarding NTP Configuration Changes 68
Releasing the CFS Session Lock 68
Verifying the NTP Configuration 69
Configuration Examples for NTP 69

CHAPTER 7 **Configuring System Message Logging** 71
Information About System Message Logging 71
Syslog Servers 72
Guidelines and Limitations for System Message Logging 72
Default Settings for System Message Logging 72
Configuring System Message Logging 73
Configuring System Message Logging to Terminal Sessions 73

Configuring System Message Logging to a File 75
Configuring Module and Facility Messages Logging 77
Configuring Logging Timestamps 79
Configuring the ACL Logging Cache 79
Applying ACL Logging to an Interface 80
Configuring a Logging Source-Interface 81
Configuring the ACL Log Match Level 82
Configuring Syslog Servers 82
Configuring syslog on a UNIX or Linux System 84
Configuring syslog Server Configuration Distribution 85
Displaying and Clearing Log Files 86
Verifying the System Message Logging Configuration 87

CHAPTER 8 **Configuring Smart Call Home 89**
Information About Smart Call Home 89
Smart Call Home Overview 90
Smart Call Home Destination Profiles 90
Smart Call Home Alert Groups 91
Smart Call Home Message Levels 92
Call Home Message Formats 93
Guidelines and Limitations for Smart Call Home 97
Prerequisites for Smart Call Home 97
Default Call Home Settings 98
Configuring Smart Call Home 98
Registering for Smart Call Home 98
Configuring Contact Information 99
Creating a Destination Profile 100
Modifying a Destination Profile 101
Associating an Alert Group with a Destination Profile 103
Adding Show Commands to an Alert Group 103
Configuring E-Mail Server Details 104
Configuring Periodic Inventory Notifications 105
Disabling Duplicate Message Throttling 106
Enabling or Disabling Smart Call Home 107

Testing the Smart Call Home Configuration 107
Verifying the Smart Call Home Configuration 108
Sample Syslog Alert Notification in Full-Text Format 109
Sample Syslog Alert Notification in XML Format 109

CHAPTER 9 **Configuring Session Manager 113**
Information About Session Manager 113
Guidelines and Limitations for Session Manager 113
Configuring Session Manager 114
Creating a Session 114
Configuring ACLs in a Session 114
Verifying a Session 115
Committing a Session 115
Saving a Session 115
Discarding a Session 115
Configuration Example for Session Manager 115
Verifying the Session Manager Configuration 116

CHAPTER 10 **Configuring the Scheduler 117**
Information About the Scheduler 117
Remote User Authentication 118
Scheduler Log Files 118
Guidelines and Limitations for the Scheduler 118
Default Settings for the Scheduler 118
Configuring the Scheduler 119
Enabling the Scheduler 119
Defining the Scheduler Log File Size 119
Configuring Remote User Authentication 120
Defining a Job 121
Deleting a Job 122
Defining a Timetable 122
Clearing the Scheduler Log File 124
Disabling the Scheduler 124
Verifying the Scheduler Configuration 125

Configuration Examples for the Scheduler 125
Creating a Scheduler Job 125
Scheduling a Scheduler Job 125
Displaying the Job Schedule 126
Displaying the Results of Running Scheduler Jobs 126
Standards for the Scheduler 126

CHAPTER 11 Configuring SNMP 127
Information About SNMP 127
SNMP Functional Overview 127
SNMP Notifications 128
SNMPv3 128
Security Models and Levels for SNMPv1, v2, and v3 128
User-Based Security Model 129
CLI and SNMP User Synchronization 130
Group-Based SNMP Access 131
Guidelines and Limitations for SNMP 131
Default SNMP Settings 131
Configuring SNMP 132
Configuring the SNMP Source Interface 132
Configuring SNMP Users 133
Enforcing SNMP Message Encryption 134
Assigning SNMPv3 Users to Multiple Roles 134
Creating SNMP Communities 134
Filtering SNMP Requests 134
Configuring SNMP Notification Receivers 135
Configuring SNMP Notification Receivers with VRFs 136
Filtering SNMP Notifications Based on a VRF 137
Configuring SNMP for Inband Access 138
Enabling SNMP Notifications 139
Configuring Link Notifications 141
Disabling Link Notifications on an Interface 141
Enabling One-Time Authentication for SNMP over TCP 142
Assigning SNMP Switch Contact and Location Information 142

Configuring the Context to Network Entity Mapping 142
Configuring the SNMP Local Engine ID 143
Disabling SNMP 144
Verifying the SNMP Configuration 144

CHAPTER 12 Using the PCAP SNMP Parser 147
Using the PCAP SNMP Parser 147

CHAPTER 13 Configuring RMON 149
Information About RMON 149
RMON Alarms 149
RMON Events 150
Configuration Guidelines and Limitations for RMON 150
Verifying the RMON Configuration 150
Default RMON Settings 151
Configuring RMON Alarms 151
Configuring RMON Events 152

CHAPTER 14 Configuring Online Diagnostics 155
Information About Online Diagnostics 155
Bootup Diagnostics 155
Health Monitoring Diagnostics 156
Expansion Module Diagnostics 157
Guidelines and Limitations for Online Diagnostics 157
Configuring Online Diagnostics 158
Verifying the Online Diagnostics Configuration 159
Default Settings for Online Diagnostics 159
Parity Error Diagnostics 159
Clearing Parity Errors 159
Soft Error Recovery 160
Verifying Memory Table Health 161

CHAPTER 15 Configuring the Embedded Event Manager 163
Information About Embedded Event Manager 163

Embedded Event Manager Policies 164
Event Statements 164
Action Statements 165
VSH Script Policies 166
Licensing Requirements for Embedded Event Manager 166
Prerequisites for Embedded Event Manager 166
Guidelines and Limitations for Embedded Event Manager 166
Default Settings for Embedded Event Manager 167
Configuring Embedded Event Manager 167
Defining an Environment Variable 167
Defining a User Policy Using the CLI 168
Configuring Event Statements 169
Configuring Action Statements 172
Defining a Policy Using a VSH Script 174
Registering and Activating a VSH Script Policy 174
Overriding a System Policy 175
Configuring Syslog as an EEM Publisher 176
Verifying the Embedded Event Manager Configuration 177
Configuration Examples for Embedded Event Manager 178
Additional References 179
Feature History for EEM 179

CHAPTER 16 **Configuring SPAN** 181
Information About SPAN 181
SPAN Sources 181
Characteristics of Source Ports 182
SPAN Destinations 182
Characteristics of Destination Ports 182
Guidelines and Limitations for SPAN 183
Creating or Deleting a SPAN Session 184
Configuring an Ethernet Destination Port 185
Configuring Source Ports 186
Configuring Source Port Channels or VLANs 187
Configuring the Description of a SPAN Session 187

Activating a SPAN Session 188
Suspending a SPAN Session 188
Displaying SPAN Information 189
Configuration Examples for SPAN 190
Configuration Example for a SPAN Session 190
Configuration Example for a Unidirectional SPAN Session 190
Configuration Example for a SPAN ACL 191
Configuration Examples for UDF-Based SPAN 191

CHAPTER 17 **Configuring Local SPAN and ERSPAN** 193
Information About ERSPAN 193
ERSPAN Sources 193
ERSPAN Destinations 194
ERSPAN Sessions 194
Multiple ERSPAN Sessions 195
High Availability 195
Prerequisites for ERSPAN 195
Guidelines and Limitations for ERSPAN 195
Default Settings for ERSPAN 199
Configuring ERSPAN 199
Configuring an ERSPAN Source Session 199
Configuring SPAN Forward Drop Traffic for ERSPAN Source Session 202
Configuring an ERSPAN ACL 203
Configuring User Defined Field (UDF) Based ACL Support 205
Configuring IPv6 User Defined Field (UDF) on ERSPAN 207
Shutting Down or Activating an ERSPAN Session 209
Verifying the ERSPAN Configuration 211
Configuration Examples for ERSPAN 211
Configuration Example for an ERSPAN Source Session 211
Configuration Example for an ERSPAN ACL 212
Configuration Examples for UDF-Based ERSPAN 212
Additional References 213
Related Documents 213

CHAPTER 18 **Configuring DNS** 215

Information About DNS Client 215

Name Servers 215

DNS Operation 215

High Availability 216

Prerequisites for DNS Clients 216

Default Settings for DNS Clients 216

Configuring the DNS Source Interface 216

Configuring DNS Clients 217

CHAPTER 19 **Configuring sFlow** 219

Information About sFlow 219

sFlow Agent 219

Prerequisites 220

Guidelines and Limitations for sFlow 220

Default Settings for sFlow 220

Configuring sFLow 220

Enabling the sFlow Feature 220

Configuring the Sampling Rate 221

Configuring the Maximum Sampled Size 221

Configuring the Counter Poll Interval 222

Configuring the Maximum Datagram Size 223

Configuring the sFlow Analyzer Address 224

Configuring the sFlow Analyzer Port 224

Configuring the sFlow Agent Address 225

Configuring the sFlow Sampling Data Source 226

Verifying the sFlow Configuration 227

Configuration Examples for sFlow 227

Additional References for sFlow 227

Feature History for sFlow 228

CHAPTER 20 **Configuring Tap Aggregation and MPLS Stripping** 229

Information About Tap Aggregation 229

Network Taps 229
Tap Aggregation 230
Guidelines and Limitations for Tap Aggregation 231
Information About MPLS Stripping 231
MPLS Overview 231
MPLS Header Stripping 231
Guidelines and Limitations for MPLS Stripping 232
Configuring Tap Aggregation 232
Enabling Tap Aggregation 232
Configuring a Tap Aggregation Policy 233
Attaching a Tap Aggregation Policy to an Interface 235
Verifying the Tap Aggregation Configuration 235
Configuring MPLS Stripping 236
Enabling MPLS Stripping 236
Adding and Deleting MPLS Labels 236
Clearing Label Entries 237
Clearing MPLS Stripping Counters 238
Configuring MPLS Label Aging 238
Configuring Destination MAC Addresses 239
Verifying the MPLS Label Configuration 239

CHAPTER 21 **Configuring Graceful Insertion and Removal** 243
About Graceful Insertion and Removal 243
Profiles 244
Snapshots 245
GIR Workflow 245
Configuring the Maintenance-Mode Profile 246
Configuring the Normal-Mode Profile 247
Creating a Snapshot 249
Adding Show Commands to Snapshots 250
Triggering Graceful Removal 252
Triggering Graceful Insertion 255
Maintenance Mode Enhancements 256
Verifying the GIR Configuration 257

CHAPTER 22 **Performing Software Maintenance Upgrades (SMUs)** 259

About SMUs 259

Package Management 260

Prerequisites for SMUs 260

Guidelines and Limitations for SMUs 261

Performing a Software Maintenance Upgrade for Cisco NX-OS 261

Preparing for Package Installation 261

Copying the Package File to a Local Storage Device or Network Server 262

Adding and Activating Packages 263

Committing the Active Package Set 264

Deactivating and Removing Packages 265

Downgrading Feature RPMs 266

Displaying Installation Log Information 267

CHAPTER 23 **Performing Configuration Replace** 269

About Configuration Replace and Commit-timeout 269

Overview 269

Benefits of Configuration Replace 271

Guidelines and Limitations for Configuration Replace 271

Recommended Workflow for Configuration Replace 272

Performing a Configuration Replace 272

Verifying Configuration Replace 274

Examples for Configuration Replace 275

CHAPTER 24 **Configuring Rollback** 279

Information About Rollbacks 279

Guidelines and Limitations for Rollbacks 279

Creating a Checkpoint 280

Implementing a Rollback 281

Verifying the Rollback Configuration 282

CHAPTER 25 **Configuring User Accounts and RBAC** 283

Information About User Accounts and RBAC 283

User Roles 283
Rules 284
User Role Policies 284
User Account Configuration Restrictions 285
User Password Requirements 285
Guidelines and Limitations for User Accounts 286
Configuring User Accounts 287
Configuring SAN Admin Users 288
Configuring RBAC 289
Creating User Roles and Rules 289
Creating Feature Groups 290
Changing User Role Interface Policies 291
Changing User Role VLAN Policies 292
Changing User Role VSAN Policies 293
Verifying the User Accounts and RBAC Configuration 293
Configuring User Accounts Default Settings for the User Accounts and RBAC 294

# Preface

This preface includes the following sections:


- Audience, on page xvii
- Document Conventions, on page xvii
- Related Documentation for Cisco Nexus 3000 Series Switches, on page xviii
- Documentation Feedback, on page xviii
- Communications, Services, and Additional Information, on page xviii


## 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. |
| *Italic* | Italic text indicates arguments for which the user supplies the values. |
| [x] | Square brackets enclose an optional element (keyword or argument). |
| [x \| y] | Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice. |
| {x \| y} | Braces enclosing keywords or arguments 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 |
|---|---|
| `variable` | Indicates a variable for which you supply values, in context where italics cannot be used. |
| string | A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks. |

Examples use the following conventions:

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

# Related Documentation for Cisco Nexus 3000 Series Switches

The entire Cisco Nexus 3000 Series switch documentation set is available at the following URL:

https://www.cisco.com/c/en/us/support/switches/nexus-3000-series-switches/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 nexus3k-docfeedback@cisco.com. We appreciate your feedback.

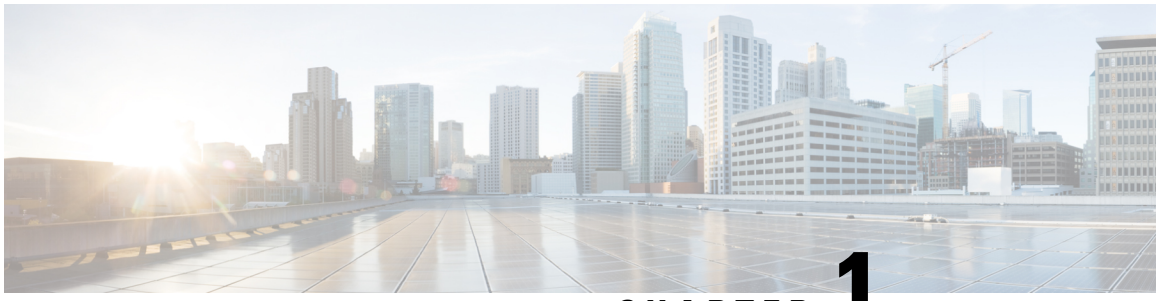
# Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business results 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 web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Preface



CHAPTER 1

# New and Changed Information

This chapter contains the following sections:

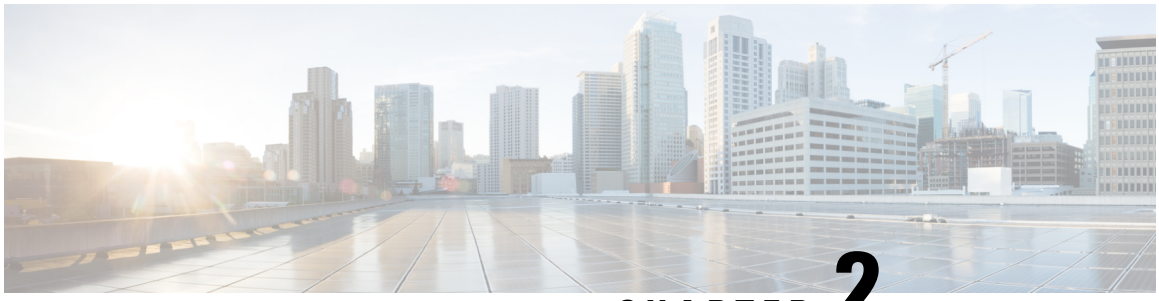

- New and Changed Information, on page 1


## New and Changed Information

The following table provides an overview of the significant changes to this guide for the current release.

| Feature | Description | Added or Changed in Release | Where Documented |
|---|---|---|---|
| Configuration replace | Introduced this feature. | 7.0(3)I7(3) | Performing Configuration Replace, on page 269 |
| ERSPAN | Added the ability to configure IPv6 User Defined Field (UDF) on ERSPAN. | 7.0(3)I6(1) | Guidelines and Limitations for ERSPAN, on page 195<br>Configuring IPv6 User Defined Field (UDF) on ERSPAN, on page 207 |
| NTP | Added the ability to scan the access group options in the following order, from least restrictive to most restrictive: peer, serve, serve-only, query-only. | 7.0(3)I6(1) | Configuring NTP Access Restrictions, on page 61 |
| SNMP | Added the ability to configure the local SNMP ID. | 7.0(3)I6(1) | Configuring the SNMP Local Engine ID, on page 143 |
| IPv6 Support for TAP Aggregation | Updated the TAP aggregation configuration section. | 7.0(3)I5(1) | Configuring a Tap Aggregation Policy, on page 233 |

| Feature | Description | Added or Changed in Release | Where Documented |
|---|---|---|---|
| GIR Maintenance Mode Enhancements | Added new maintenance mode enhancements for GIR. | 7.0(3)I5(1) | Maintenance Mode Enhancements, on page 256 |
| SNMP | Updated the MIBs information. | 7.0(3)I4(2) | Enabling SNMP Notifications, on page 139 |
| ERSPAN and SPAN | Added support for the same source in multiple sessions and for multiple ACL filters on the same source. | 7.0(3)I4(1) | Guidelines and Limitations for SPAN, on page 183<br>Guidelines and Limitations for ERSPAN, on page 195 |



CHAPTER 2

# Overview

This chapter contains the following sections:


- Licensing Requirements, on page 3
- System Management Features, on page 3


## 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*.

## System Management Features

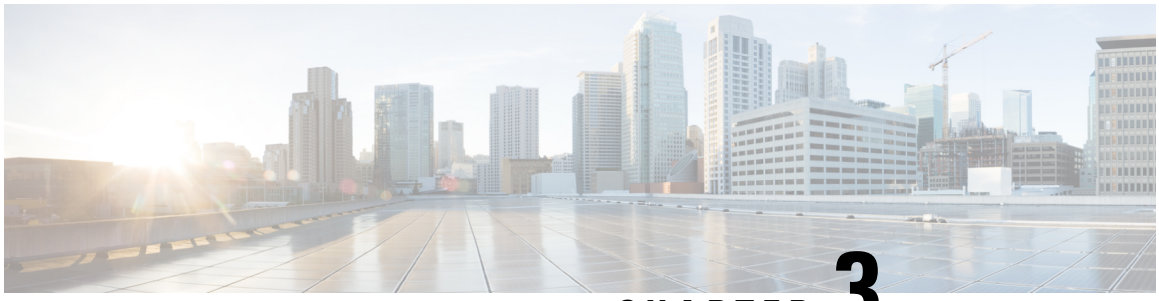
The system management features documented in this guide are described below:

| Feature | Description |
|---|---|
| Switch Profiles | Configuration synchronization allows administrators to make configuration changes on one switch and have the system automatically synchronize the configuration to a peer switch. This feature eliminates misconfigurations and reduces the administrative overhead.<br><br>The configuration synchronization mode (config-sync) allows users to create switch profiles to synchronize local and peer switch. |
| Cisco Fabric Services | The Cisco MDS NX-OS software uses the Cisco Fabric Services (CFS) infrastructure to enable efficient database distribution and to promote device flexibility. CFS simplifies SAN provisioning by automatically distributing configuration information to all switches in a fabric. |

| Feature | Description |
|---|---|
| Precision Time Protocol | The Precision Time Protocol (PTP) is a time synchronization protocol for nodes distributed across a network. Its hardware timestamp feature provides greater accuracy than other time synchronization protocols such as Network Time Protocol (NTP). |
| User Accounts and RBAC | User accounts and role-based access control (RBAC) allow you to define the rules for an assigned role. Roles restrict the authorization that the user has to access management operations. Each user role can contain multiple rules and each user can have multiple roles. |
| 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. |
| 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.<br><br>The platform-specific diagnostics provide hardware-specific fault-detection tests and allow you to take appropriate corrective action in response to diagnostic test results. |
| 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 a terminal session, a log file, and syslog servers on remote systems.<br><br>System message logging is based on RFC 3164. For more information about the system message format and the messages that the device generates, see the *Cisco NX-OS System Messages Reference*. |
| Smart Call Home | Call Home provides an e-mail-based notification of critical system policies. Cisco NX-OS provides a range of message formats for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications. You can use this feature to page a network support engineer, e-mail a Network Operations Center, or use Cisco Smart Call Home services to automatically generate a case with the Technical Assistance Center. |

| Feature | Description |
|---|---|
| Configuration Rollback | The configuration rollback feature allows users to take a snapshot, or user checkpoint, of the Cisco NX-OS configuration and then reapply that configuration to a switch at any point without having to reload the switch. A rollback allows any authorized administrator to apply this checkpoint configuration without requiring expert knowledge of the features configured in the checkpoint. |
| 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. |
| RMON | RMON is an 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. |
| SPAN | The Switched Port Analyzer (SPAN) feature (sometimes called port mirroring or port monitoring) selects network traffic for analysis by a network analyzer. The network analyzer can be a Cisco SwitchProbe, a Fibre Channel Analyzer, or other Remote Monitoring (RMON) probes. |

| Feature | Description |
| --- | --- |
| 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. ERSPAN uses a generic routing encapsulation (GRE) tunnel to carry traffic between switches.<br><br>ERSPAN consists of an ERSPAN source session, routable ERSPAN GRE-encapsulated traffic, and an ERSPAN destination session. You separately configure ERSPAN source sessions and destination sessions on different switches.<br><br>To configure an ERSPAN source session on one switch, you associate a set of source ports or VLANs with a destination IP address, ERSPAN ID number, and virtual routing and forwarding (VRF) name. To configure an ERSPAN destination session on another switch, you associate the destinations with the source IP address, the ERSPAN ID number, and a VRF name.<br><br>The ERSPAN source session copies traffic from the source ports or source VLANs and forwards the traffic using routable GRE-encapsulated packets to the ERSPAN destination session. The ERSPAN destination session switches the traffic to the destinations. |



CHAPTER 3

# Configuring Switch Profiles

This chapter contains the following sections:


- Information About Switch Profiles, on page 7
- Switch Profile Configuration Modes, on page 8
- Configuration Validation, on page 8
- Software Upgrades and Downgrades with Switch Profiles, on page 9
- Prerequisites for Switch Profiles, on page 10
- Guidelines and Limitations for Switch Profiles, on page 10
- Configuring Switch Profiles, on page 11
- Adding a Switch to a Switch Profile, on page 13
- Adding or Modifying Switch Profile Commands, on page 14
- Importing a Switch Profile, on page 17
- Verifying Commands in a Switch Profile, on page 19
- Isolating a Peer Switch, on page 19
- Deleting a Switch Profile, on page 20
- Deleting a Switch from a Switch Profile, on page 20
- Displaying the Switch Profile Buffer, on page 21
- Synchronizing Configurations After a Switch Reboot, on page 22
- Switch Profile Configuration show Commands, on page 23
- Supported Switch Profile Commands, on page 23
- Configuration Examples for Switch Profiles, on page 25


# Information About Switch Profiles

Cisco NX-OS Release 6.0(2)U4(1) introduces Switch Profiles. Several applications require consistent configuration across Cisco Nexus Series switches in the network. 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.

# Switch Profile Configuration Modes

The switch profile feature includes the following configuration modes:

- Configuration Synchronization Mode
- Switch Profile Mode
- Switch Profile Import Mode

### Configuration Synchronization Mode

The configuration synchronization mode (config-sync) allows you to create switch profiles using the **config sync** command on the local switch that you want to use as the primary. After you create the profile, you can enter the **config sync** command on the peer switch that you want to synchronize.

### Switch Profile Mode

The switch profile mode allows you to add supported configuration commands to a switch profile that is later synchronized with a peer switch. Commands that you enter in the switch profile mode are buffered until you enter the **commit** command.

### Switch Profile Import Mode

When you upgrade from an earlier release, you have the option to enter the **import** command to copy supported running-configuration commands to a switch profile. After entering the **import** command, the switch profile mode (config-sync-sp) changes to the switch profile import mode (config-sync-sp-import). The switch profile import mode allows you to import existing switch configurations from the running configuration and specify which commands you want to include in the switch profile.

Because different topologies require different commands that are included in a switch profile, the **import** command mode allows you to modify the imported set of commands to suit a specific topology.

You need to enter the **commit** command to complete the import process and move the configuration into the switch profile. Because configuration changes are not supported during the import process, if you added new commands before entering the **commit** command, the switch profile remains unsaved and the switch remains in the switch profile import mode. You can remove the added commands or abort the import. Unsaved configurations are lost if the process is aborted. You can add new commands to the switch profile after the import is complete.

# Configuration Validation

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

- Mutual Exclusion Checks

- Merge Checks

### Mutual Exclusion Checks

To reduce the possibility of overriding configuration settings that are included in a switch profile, mutual exclusion (mutex) checks the switch profile commands against the commands that exist on the local switch and the commands on the peer switch. A command that is included in a switch profile cannot be configured outside of the switch profile or on a peer switch. This requirement reduces the possibility that an existing command is unintentionally overwritten.

As a part of the commit process, the mutex-check occurs on both switches if the peer switch is reachable; otherwise, the mutex-check is performed locally. Configuration changes made from the configuration terminal occur only on the local switch.

If a mutex-check identifies errors, they are reported as mutex failures and they must be manually corrected.

The following exceptions apply to the mutual exclusion policy:

- Interface configuration—Port channel interfaces must be configured fully in either switch profile mode or global configuration mode.

**Note** Several port channel subcommands are not configurable in switch profile mode. These commands can be configured from global configuration mode even if the port channel is created and configured in switch profile mode.

For example, the following command can only be configured in global configuration mode:

**switchport private-vlan association trunk** *primary-vlan secondary-vlan*

- 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 merge or commit process. Errors are reported as merge failures and must be manually corrected.

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

When you downgrade to an earlier release, you are prompted to remove an existing switch profile that is not supported on earlier releases.

When you upgrade from an earlier release, you have the option to move some of the running-configuration commands to a switch profile. The **import** command allows you to import relevant switch profile commands.

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

When you perform an In Service Software Upgrade (ISSU) on one of the switches included in a switch profile, a configuration synchronization cannot occur because the peer is unreachable.

# Prerequisites for Switch Profiles

Switch profiles have the following prerequisites:

- You must enable Cisco Fabric Series over IP (CFSoIP) distribution over mgmt0 on both switches by entering the **cfs ipv4 distribute** command.
- You must configure a switch profile with the same name on both peer switches by entering the **config sync** and **switch-profile** commands.
- Configure each switch as peer switch by entering the **sync-peers destination** command

# Guidelines and Limitations for Switch Profiles

Consider the following guidelines and limitations when configuring switch profiles:

- You can only enable configuration synchronization using the mgmt0 interface.
- Configuration synchronization is performed using the mgmt 0 interface and cannot be performed using a management SVI.
- 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 (config-sync-sp) mode.
- One switch profile session can be in progress at a time. Attempts to start another session will fail.
- Supported command changes made from the configuration terminal mode are blocked when a switch profile session is in progress. You should not make unsupported command changes from the configuration terminal mode 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 there is a commit failure, the commands remain in the switch profile buffer. You can then make necessary corrections and try the commit again.
- Once a port channel is configured using switch profile mode, it cannot be configured using global configuration (config terminal) mode.

**Note** Several port channel sub-commands are not configurable in switch profile mode. These commands can be configured from global configuration mode even if the port channel is created and configured in switch profile mode.

For example, the following command can only be configured in global configuration mode:

**switchport private-vlan association trunk** *primary-vlan secondary-vlan*

- Shutdown and no shutdown can be configured in either global configuration mode or switch profile 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 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 including the member interface must also be present within the switch profile.

### Guidelines for Synchronizing After Connectivity Loss

- Synchronizing configurations after mgmt0 interface connectivity loss—When 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 synchronize automatically.

  If a configuration change is made on only one switch, a merge will occur when the mgmt0 interface comes up and the configuration is applied on the other switch.

# Configuring Switch Profiles

You can create and configure a switch profile. Enter the **switch-profile** *name* command in the configuration synchronization mode (config-sync).

### Before you begin

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

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch# configure terminal`<br>`switch(config)#` | |
| **Step 2** | **cfs ipv4 distribute**<br>**Example:**<br>`switch(config)# cfs ipv4 distribute`<br>`switch(config)#` | Enables CFS distribution between the peer switches. |
| **Step 3** | **config sync**<br>**Example:**<br>`switch# config sync`<br>`switch(config-sync)#` | Enters configuration synchronization mode. |
| **Step 4** | **switch-profile** *name*<br>**Example:**<br>`switch(config-sync)# switch-profile abc`<br>`switch(config-sync-sp)#` | Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode. |
| **Step 5** | **sync-peers destination** *IP-address*<br>**Example:**<br>`switch(config-sync-sp)# sync-peers destination 10.1.1.1`<br>`switch(config-sync-sp)#` | Configures the peer switch. |
| **Step 6** | (Optional) **show switch-profile** *name* **status**<br>**Example:**<br>`switch(config-sync-sp)# show switch-profile abc status`<br>`switch(config-sync-sp)#` | Views the switch profile on the local switch and the peer switch information. |
| **Step 7** | **exit**<br>**Example:**<br>`switch(config-sync-sp)# exit`<br>`switch#` | Exits the switch profile configuration mode and returns to EXEC mode. |
| **Step 8** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure a switch profile and shows the switch profile status.

```
switch# configuration terminal
switch(config)# cfs ipv4 distribute
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# sync-peers destination 10.1.1.1
switch(config-sync-sp)# show switch-profile abc status
```

```
Start-time:  15801 usecs after Mon Aug 23 06:21:08 2010
End-time:   6480 usecs after Mon Aug 23 06:21:13 2010

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: 10.1.1.1
Sync-status: In Sync.
Status: Commit Success
Error(s):
switch(config-sync-sp)# exit
switch#
```

# Adding a Switch to a Switch Profile

Enter the **sync-peers destination** *destination IP* command in switch profile configuration mode to add the switch to a switch profile.

Follow these guidelines when adding switches:

- Switches are identified by their IP address.
- Destination IPs are the IP addresses of the switches that you want to synchronize.
- The committed switch profile is synchronized with the newly added peers (when they are online) if the peer switch is also configured with configuration synchronization.

  If you want to import a member interface to a switch profile, the port channel including the member interface must also be present within the switch profile.

### Before you begin

After creating a switch profile on the local switch, you must add the second switch that will be included in the synchronization.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **config sync**<br>**Example:**<br>`switch# config sync`<br>`switch(config-sync)#` | Enters configuration synchronization mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | **switch-profile** *name*<br>**Example:**<br>`switch(config-sync)# switch-profile abc`<br>`switch(config-sync-sp)#` | Configures switch profile, names the switch profile, and enters switch profile synchronization configuration mode. |
| **Step 3** | **sync-peers destination** *destination IP*<br>**Example:**<br>`switch(config-sync-sp)# sync-peers destination 10.1.1.1`<br>`switch(config-sync-sp)#` | Adds a switch to the switch profile. |
| **Step 4** | **exit**<br>**Example:**<br>`switch(config-sync-sp)# exit`<br>`switch#` | Exits switch profile configuration mode. |
| **Step 5** | (Optional) **show switch-profile peer**<br>**Example:**<br>`switch# show switch-profile peer` | Displays the switch profile peer configuration. |
| **Step 6** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

# Adding or Modifying Switch Profile Commands

To modify a command in a switch profile, add the modified command to the switch profile and enter the **commit** command to apply the command and synchronize the switch profile to the peer switch if it is reachable.

Follow these guidelines when adding or modifying switch profile commands:

- 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, or the **buffer-move** command, to change the buffer and correct the order of already entered commands.

### Before you begin

After configuring a switch profile on the local and the peer switch, you must add and commit the supported commands to the switch profile. The commands are added to the switch profile buffer until you enter the **commit** command. The **commit** command does the following:

- Triggers the mutex check and the merge check to verify the synchronization.

- Creates a checkpoint with a rollback infrastructure.
- Applies the configuration on the local switch and the peer switch.
- Executes a rollback on all switches if there is a failure with an application on any of the switches in the switch profile.
- Deletes the checkpoint.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **config sync**<br>**Example:**<br>`switch# config sync`<br>`switch(config-sync)#` | Enters configuration synchronization mode. |
| **Step 2** | **switch-profile** *name*<br>**Example:**<br>`switch(config-sync)# switch-profile abc`<br>`switch(config-sync-sp)#` | Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode. |
| **Step 3** | *Command argument*<br>**Example:**<br>`switch(config-sync-sp)# interface Port-channel100`<br>`switch(config-sync-sp-if)# speed 1000`<br>`switch(config-sync-sp-if)# interface Ethernet1/1`<br>`switch(config-sync-sp-if)# speed 1000`<br>`switch(config-sync-sp-if)# channel-group 100` | Adds a command to the switch profile. |
| **Step 4** | (Optional) **show switch-profile** *name* **buffer**<br>**Example:**<br>`switch(config-sync-sp)# show switch-profile abc buffer`<br>`switch(config-sync-sp)#` | Displays the configuration commands in the switch profile buffer. |
| **Step 5** | **verify**<br>**Example:**<br>`switch(config-sync-sp)# verify` | Verifies the commands in the switch profile buffer. |
| **Step 6** | **commit**<br>**Example:**<br>`switch(config-sync-sp)# commit` | Saves the commands in the switch profile and synchronizes the configuration with the peer switch. |
| **Step 7** | (Optional) **show switch-profile** *name* **status**<br>**Example:** | Displays the status of the switch profile on the local switch and the status on the peer switch. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch(config-sync-sp)# show switch-profile abc status`<br>`switch(config-sync-sp)#` | |
| **Step 8** | **exit**<br>**Example:**<br>`switch(config-sync-sp)# exit`<br>`switch#` | Exits the switch profile configuration mode. |
| **Step 9** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to create a switch profile, configure a peer switch, and add commands to the switch profile.

```
switch# configuration terminal
switch(config)# cfs ipv4 distribute
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# sync-peers destination 10.1.1.1
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
switch(config-sync-sp-if)# channel-group 100
switch(config-sync-sp)# verify
switch(config-sync-sp)# commit
switch(config-sync-sp)# exit
switch#
```

The following example shows an existing configuration with a defined switch profile. The second example shows how the switch profile command changed by adding the modified command to the switch profile.

```
switch# show running-config
switch-profile abc
  interface Ethernet1/1
    switchport mode trunk
    switchport trunk allowed vlan 1-10

switch# config sync
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# interface Ethernet1/1
switch(config-sync-sp-if)# switchport trunk allowed vlan 5-10
switch(config-sync-sp-if)# commit

switch# show running-config
switch-profile abc
  interface Ethernet1/1
    switchport mode trunk
    switchport trunk allowed vlan 5-10
```

# Importing a Switch Profile

You can import a switch profile based on the set of commands that you want to import. Using the configuration terminal mode, you can do the following:

- Add selected commands to the switch profile.
- Add supported commands that were specified for an interface.
- Add supported system-level commands.
- Add supported system-level commands excluding the physical interface commands.

When you import commands to a switch profile, the switch profile buffer must be empty.

If new commands are added during the import, the switch profile remains unsaved and the switch remains in the switch profile import mode. You can enter the **abort** command to stop the import. For additional information importing a switch profile, see the "Switch Profile Import Mode" section.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **config sync**<br>**Example:**<br>`switch# config sync`<br>`switch(config-sync)#` | Enters configuration synchronization mode. |
| **Step 2** | **switch-profile** *name*<br>**Example:**<br>`switch(config-sync)# switch-profile abc`<br>`switch(config-sync-sp)#` | Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode. |
| **Step 3** | **import** {*interface port/slot* \| *running-config* [**exclude interface ethernet**]}<br>**Example:**<br>`switch(config-sync-sp)# import ethernet 1/2`<br>`switch(config-sync-sp-import)#` | Identifies the commands that you want to import and enters switch profile import mode.<br>• <CR>—Adds selected commands.<br>• **interface**—Adds the supported commands for a specified interface.<br>• **running-config**—Adds supported system-level commands.<br>• **running-config exclude interface ethernet**—Adds supported system-level commands excluding the physical interface commands. |
| **Step 4** | **commit**<br>**Example:**<br>`switch(config-sync-sp-import)# commit` | Imports the commands and saves the commands to the switch profile. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 5** | (Optional) **abort**<br>**Example:**<br>`switch(config-sync-sp-import)# abort` | Aborts the import process. |
| **Step 6** | **exit**<br>**Example:**<br>`switch(config-sync-sp)# exit`<br>`switch#` | Exits switch profile import mode. |
| **Step 7** | (Optional) **show switch-profile**<br>**Example:**<br>`switch# show switch-profile` | Displays the switch profile configuration. |
| **Step 8** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to import supported system-level commands excluding the Ethernet interface commands into the switch profile named sp:

```
switch(config-vlan)# conf sync
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# show switch-profile buffer

switch-profile  : sp
----------------------------------------------------------
Seq-no  Command
----------------------------------------------------------

switch(config-sync-sp)# import running-config exclude interface ethernet
switch(config-sync-sp-import)#
switch(config-sync-sp-import)# show switch-profile buffer

switch-profile  : sp
----------------------------------------------------------
Seq-no  Command
----------------------------------------------------------
3       vlan 100-299
4       vlan 300
4.1       state suspend
5       vlan 301-345
6       interface port-channel100
6.1       spanning-tree port type network
7       interface port-channel105

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

# Verifying Commands in a Switch Profile

You can verify the commands that are included in a switch profile by entering the **verify** command in switch profile mode.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **config sync**<br>**Example:**<br>`switch# config sync`<br>`switch(config-sync)#` | Enters configuration synchronization mode. |
| **Step 2** | **switch-profile** *name*<br>**Example:**<br>`switch(config-sync)# switch-profile abc`<br>`switch(config-sync-sp)#` | Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode. |
| **Step 3** | **verify**<br>**Example:**<br>`switch(config-sync-sp)# verify` | Verifies the commands in the switch profile buffer. |
| **Step 4** | **exit**<br>**Example:**<br>`switch(config-sync-sp)# exit`<br>`switch#` | Exits the switch profile configuration mode. |
| **Step 5** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

# 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 a configuration synchronization or when you want to debug configurations.

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

To temporarily isolate a peer switch, follow these steps:

1. Remove a peer switch from a switch profile.
2. Make changes to the switch profile and commit the changes.
3. Enter debug commands.

4. Undo the changes that were made to the switch profile in Step 2 and commit.
5. Add the peer switch back to the switch profile.

# Deleting a Switch Profile

You can delete a switch profile by selecting the **all-config** or the **local-config** option:

- **all-config**—Deletes the switch profile on both peer switches (when both are reachable). If you choose this option and one of the peers is unreachable, only the local switch profile is deleted. The **all-config** option completely deletes the switch profile on both peer switches.
- **local-config**—Deletes the switch profile on the local switch only.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **config sync**<br>**Example:**<br>`switch# config sync`<br>`switch(config-sync)#` | Enters configuration synchronization mode. |
| **Step 2** | **no switch-profile name** {**all-config** \| **local-config**}<br>**Example:**<br>`switch(config-sync)# no switch-profile abc local-config`<br>`switch(config-sync-sp)#` | Deletes the switch profile as follows:<br>• **all-config**—Deletes the switch profile on the local and peer switch. If the peer switch is not reachable, only the local switch profile is deleted.<br>• **local-config**—Deletes the switch profile and local configuration. |
| **Step 3** | **exit**<br>**Example:**<br>`switch(config-sync-sp)# exit`<br>`switch#` | Exits configuration synchronization mode. |
| **Step 4** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

# Deleting a Switch from a Switch Profile

You can delete a switch from a switch profile.

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **config sync**<br>**Example:**<br>`switch# config sync`<br>`switch(config-sync)#` | Enters configuration synchronization mode. |
| **Step 2** | **switch-profile** *name*<br>**Example:**<br>`switch(config-sync)# switch-profile abc`<br>`switch(config-sync-sp)#` | Configures the switch profile, names the switch profile, and enters the switch profile synchronization configuration mode. |
| **Step 3** | **no sync-peers destination** *destination IP*<br>**Example:**<br>`switch(config-sync-sp)# no sync-peers destination 10.1.1.1`<br>`switch(config-sync-sp)#` | Removes the specified switch from the switch profile. |
| **Step 4** | **exit**<br>**Example:**<br>`switch(config-sync-sp)# exit`<br>`switch#` | Exits the switch profile configuration mode. |
| **Step 5** | (Optional) **show switch-profile**<br>**Example:**<br>`switch# show switch-profile` | Displays the switch profile configuration. |
| **Step 6** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

# Displaying the Switch Profile Buffer

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure sync** | Enters configuration synchronization mode. |
| **Step 2** | switch(config-sync) # **switch-profile** *profile-name* | Enters switch profile synchronization configuration mode for the specified switch profile. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-sync-sp) # **show switch-profile***profile-name* **buffer** | Enters interface switch profile synchronization configuration mode for the specified interface. |

### Example

The following example shows how to display the switch profile buffer for a service profile called sp:

```
switch# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(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

switch(config-sync-sp)# buffer-move 3 1
switch(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
switch(config-sync-sp)#
```

# Synchronizing Configurations After a Switch Reboot

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

### Procedure

**Step 1** Reapply configurations that were changed on the peer switch during the reboot.

**Step 2** Enter the **commit** command.

**Step 3** Verify that the configuration is applied correctly and both peers are back synchronized.

**Example**

# Switch Profile Configuration show Commands

The following **show** commands display information about the switch profile.

| Command | Purpose |
|---|---|
| **show switch-profile** *name* | Displays the commands in a switch profile. |
| **show switch-profile** *name* **buffer** | Displays the uncommitted commands in a switch profile, the commands that were moved, and the commands that were deleted. |
| **show switch-profile** *name* **peer** *IP-address* | Displays the synchronization status for a peer switch. |
| show switch-profile *name* **session-history** | Displays the status of the last 20 switch profile sessions. |
| **show switch-profile** *name* **status** | Displays the configuration synchronization status of a peer switch. |
| **show running-config exclude-provision** | Displays the configurations for offline preprovisioned interfaces that are hidden. |
| **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. |

For detailed information about the fields in the output from these commands, see the system management command reference for your platform.

# Supported Switch Profile Commands

The following switch profile commands are supported:

- **logging event link-status default**
- [**no**] **vlan** *vlan-range*
- **ip access-list** *acl-name*
- **policy-map type network-qos jumbo-frames**
  - **class type network-qos class-default**
  - **mtu** *mtu value*
- **system qos**
  - **service-policy type network-qos jumbo-frames**

- **vlan configuration** *vlan id*
  - **ip igmp snooping querier** *ip*
- **spanning-tree port type edge default**
- **spanning-tree port type edge bpduguard default**
- **spanning-tree loopguard default**
- **no spanning-tree vlan** *vlan id*
- **port-channel load-balance ethernet source-dest-port**
- **interface port-channel** *number*
  - **description** *text*
  - **switchport mode trunk**
  - **switchport trunk allowed vlan** *vlan list*
  - **spanning-tree port type network**
  - **no negotiate auto**
  - **vpc peer-link**
- **interface port-channel** *number*
  - **switchport access vlan** *vlan id*
  - **spanning-tree port type edge**
  - **speed 10000**
  - **vpc** *number*
- **interface ethernet***x/y*
  - **switchport access vlan** *vlanid*
  - **spanning-tree port type edge**
  - **channel-group** *number* **mode active**
- **service dhcp**
- **ip dhcp relay**
- **ipv6 dhcp relay**
- **storm-control unicast level**

# Configuration Examples for Switch Profiles

## Creating a Switch Profile on a Local and Peer Switch Example

The following example shows how to create a successful switch profile configuration on a local and peer switch.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Enable CFSoIP distribution on the local and the peer switch.<br>**Example:**<br>`switch# configuration terminal`<br>`switch(config)# cfs ipv4 distribute` | |
| **Step 2** | Create a switch profile on the local and the peer switch.<br>**Example:**<br>`switch(config-sync)# switch-profile abc`<br><br>`switch(config-sync-sp)# sync-peers destination 10.1.1.1` | |
| **Step 3** | Verify that the switch profiles are the same on the local and the peer switch.<br>**Example:**<br>`switch(config-sync-sp)# show switch-profile abc status`<br><br>`Start-time: 15801 usecs after Mon Aug 23 06:21:08 2010`<br>`End-time: 6480 usecs after Mon Aug 23 06:21:13 2010`<br><br>`Profile-Revision: 1`<br>`Session-type: Initial-Exchange`<br>`Peer-triggered: Yes`<br>`Profile-status: Sync Success`<br><br>`Local information:`<br>`----------`<br>`Status: Commit Success`<br>`Error(s):`<br><br>`Peer information:`<br>`----------`<br>`IP-address: 10.1.1.1`<br>`Sync-status: In Sync.`<br>`Status: Commit Success`<br>`Error(s):` | |

| | Command or Action | Purpose |
|---|---|---|
| **Step 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.<br>**Example:**<br>`switch(config-sync-sp)# class-map type qos c1` | |
| **Step 5** | Verify the commands in the switch profile.<br>**Example:**<br>`switch(config-sync-sp-if)# verify`<br>`Verification Successful` | |
| **Step 6** | Apply the commands to the switch profile and to synchronize the configurations between the local and the peer switch.<br>**Example:**<br>`switch(config-sync-sp)# commit`<br>`Commit Successful`<br>`switch(config-sync)#` | |

# Verifying the Synchronization Status Example

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

```
switch(config-sync)# show switch-profile switch-profile status
Start-time: 804935 usecs after Mon Aug 23 06:41:10 2010
End-time: 956631 usecs after Mon Aug 23 06:41:20 2010

Profile-Revision: 2
Session-type: Commit
Peer-triggered: No
Profile-status: Sync Success

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

Peer information:
----------------
IP-address: 10.1.1.1
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch(config-sync)#
```

# Displaying the Running Configuration

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

```
switch# configure sync
switch(config-sync)# show running-config switch-profile

switch(config-sync)#
```

# Displaying the Switch Profile Synchronization Between Local and Peer Switches

This example shows how to display the synchronization status for two peer switches:

```
switch1# show switch-profile sp status

Start-time: 491815 usecs after Thu Aug 12 11:54:51 2010
End-time: 449475 usecs after Thu Aug 12 11:54:58 2010

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):

switch1#


switch2# show switch-profile sp status

Start-time: 503194 usecs after Thu Aug 12 11:54:51 2010
End-time: 532989 usecs after Thu Aug 12 11:54:58 2010

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):
```

```
switch2#
```

# Displaying Verify and Commit on Local and Peer Switches

This example shows how to configure a successful verify and commit of the local and peer switch:

```
switch1# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
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)# 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 Aug 11 17:51:28 2010
End-time: 676451 usecs after Wed Aug 11 17:51:43 2010

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 Wed Aug 11 16:51:28 2010
End-time: 734702 usecs after Wed Aug 11 16:51:43 2010

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):

switch2#
```

# Successful and Unsuccessful Synchronization Examples

The following example shows a successful synchronization of the switch profile on the peer switch:

```
switch# show switch-profile abc peer

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

The following example shows an unsuccessful synchronization of a switch profile on the peer switch, with a peer not reachable status:

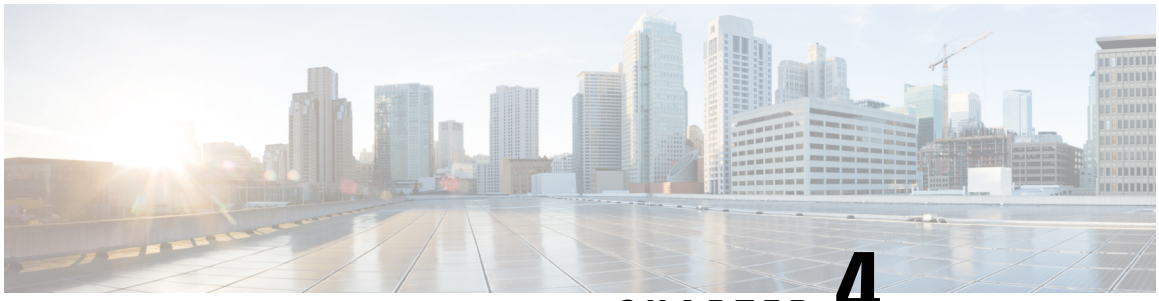
```
switch# show switch-profile sp peer 10.193.194.52
Peer-sync-status          : Not yet merged. pending-merge:1 received_merge:0
Peer-status               : Peer not reachable
Peer-error(s)             :
switch#
```

# Configuring the Switch Profile Buffer, Moving the Buffer, and Deleting the Buffer

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

```
switch# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# vlan 101
switch(config-sync-sp-vlan)# ip igmp snooping querier 10.101.1.1
switch(config-sync-sp-vlan)# exit
switch(config-sync-sp)# mac address-table static 0000.0000.0001 vlan 101 drop
switch(config-sync-sp)# interface ethernet1/2
switch(config-sync-sp-if)# switchport mode trunk
switch(config-sync-sp-if)# switchport trunk allowed vlan 101
switch(config-sync-sp-if)# exit
switch(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

switch(config-sync-sp)# buffer-move 3 1
switch(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

switch(config-sync-sp)# buffer-delete 1
switch(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

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



CHAPTER 4

# Using Cisco Fabric Services

This chapter contains the following sections:


- Information About CFS, on page 31
- CFS Distribution, on page 32
- CFS Support for Applications, on page 33
- CFS Regions, on page 36
- Configuring CFS over IP, on page 39
- Default Settings for CFS, on page 40


## Information About CFS

Some features in the Cisco Nexus Series switch require configuration synchronization with other switches in the network to function correctly. Synchronization through manual configuration at each switch in the network can be a tedious and error-prone process.

Cisco Fabric Services (CFS) provides a common infrastructure for automatic configuration synchronization in the network. It provides the transport function and a set of common services to the features. CFS has the ability to discover CFS-capable switches in the network and to discover feature capabilities in all CFS-capable switches.

Cisco Nexus Series switches support CFS message distribution over IPv4 or IPv6 networks.

CFS provides the following features:

- Peer-to-peer protocol with no client-server relationship at the CFS layer.
- CFS message distribution over IPv4 networks.
- Three modes of distribution.
  - Coordinated distributions—Only one distribution is allowed in the network at any given time.
  - Uncoordinated distributions—Multiple parallel distributions are allowed in the network except when a coordinated distribution is in progress.
  - Unrestricted uncoordinated distributions—Multiple parallel distributions are allowed in the network in the presence of an existing coordinated distribution. Unrestricted uncoordinated distributions are allowed to run in parallel with all other types of distributions.

The following features are supported for CFS distribution over IP:

- One scope of distribution over an IP network:
  - Physical scope—The distribution spans the entire IP network.

# CFS Distribution

The CFS distribution functionality is independent of the lower layer transport. Cisco Nexus Series switches support CFS distribution over IP. Features that use CFS are unaware of the lower layer transport.

# CFS Distribution Modes

CFS supports three distribution modes to accommodate different feature requirements:

- Uncoordinated Distribution
- Coordinated Distribution
- Unrestricted Uncoordinated Distributions

Only one mode is allowed at any given time.

## Uncoordinated Distribution

Uncoordinated distributions are used to distribute information that is not expected to conflict with information from a peer. Parallel uncoordinated distributions are allowed for a feature.

## Coordinated Distribution

Coordinated distributions allow only one feature distribution at a given time. CFS uses locks to enforce this feature. A coordinated distribution is not allowed to start if locks are taken for the feature anywhere in the network. A coordinated distribution consists of three stages:

- A network lock is acquired.
- The configuration is distributed and committed.
- The network lock is released.

Coordinated distribution has two variants:

- CFS driven —The stages are executed by CFS in response to a feature request without intervention from the feature.
- Feature driven—The stages are under the complete control of the feature.

Coordinated distributions are used to distribute information that can be manipulated and distributed from multiple switches, for example, the port security configuration.

## Unrestricted Uncoordinated Distributions

Unrestricted uncoordinated distributions allow multiple parallel distributions in the network in the presence of an existing coordinated distribution. Unrestricted uncoordinated distributions are allowed to run in parallel with all other types of distributions.

## Verifying the CFS Distribution Status

The **show cfs status** command displays the status of CFS distribution on the switch:

```
switch# show cfs status
Distribution : Enabled
Distribution over IP : Enabled - mode IPv4
IPv4 multicast address : 239.255.70.83

Distribution over Ethernet : Enabled
```

# CFS Support for Applications

## CFS Application Requirements

All switches in the network must be CFS capable. Switches that are not CFS capable do not receive distributions, which results in part of the network not receiving the intended distribution. CFS has the following requirements:

- Implicit CFS usage—The first time that you issue a CFS task for a CFS-enabled application, the configuration modification process begins and the application locks the network.
- Pending database—The pending database is a temporary buffer to hold uncommitted information. The uncommitted changes are not applied immediately to ensure that the database is synchronized with the database in the other switches in the network. When you commit the changes, the pending database overwrites the configuration database (also known as the active database or the effective database).
- CFS distribution enabled or disabled on a per-application basis—The default (enable or disable) for the CFS distribution state differs between applications. If CFS distribution is disabled for an application, that application does not distribute any configuration and does not accept a distribution from other switches in the network.
- Explicit CFS commit—Most applications require an explicit commit operation to copy the changes in the temporary buffer to the application database, to distribute the new database to the network, and to release the network lock. The changes in the temporary buffer are not applied if you do not perform the commit operation.

## Enabling CFS for an Application

All CFS-based applications provide an option to enable or disable the distribution capabilities.

Applications have the distribution enabled by default.

The application configuration is not distributed by CFS unless distribution is explicitly enabled for that application.

### Verifying Application Registration Status

The **show cfs application** command displays the applications that are currently registered with CFS. The first column displays the application name. The second column indicates whether the application is enabled or disabled for distribution (enabled or disabled). The last column indicates the scope of distribution for the application (logical, physical, or both).

**Note** The **show cfs application** command only displays applications registered with CFS. Conditional services that use CFS do not appear in the output unless these services are running.

```
switch# show cfs application

----------------------------------------------
 Application    Enabled   Scope
----------------------------------------------
 ntp            No        Physical-all
 fscm           Yes       Physical-fc
 rscn           No        Logical
 fctimer        No        Physical-fc
 syslogd        No        Physical-all
 callhome       No        Physical-all
 fcdomain       Yes       Logical
 device-alias   Yes       Physical-fc
Total number of entries = 8
```

The **show cfs application name** command displays the details for a particular application. It displays the enabled/disabled state, timeout as registered with CFS, merge capability (if it has registered with CFS for merge support), and the distribution scope.

```
switch# show cfs application name fscm

Enabled         : Yes
 Timeout        : 100s
 Merge Capable  : No
 Scope          : Physical-fc
```

# Locking the Network

When you configure (first-time configuration) a feature (application) that uses the CFS infrastructure, that feature starts a CFS session and locks the network. When a network is locked, the switch software allows configuration changes to this feature only from the switch that holds the lock. If you make configuration changes to the feature from another switch, the switch issues a message to inform the user about the locked status. The configuration changes are held in a pending database by that application.

If you start a CFS session that requires a network lock but forget to end the session, an administrator can clear the session. If you lock a network at any time, your username is remembered across restarts and switchovers. If another user (on the same machine) tries to perform configuration tasks, that user's attempts are rejected.

### Verifying CFS Lock Status

The **show cfs lock** command displays all the locks that are currently acquired by any application. For each application the command displays the application name and scope of the lock taken.

The **show cfs lock name** command displays the lock details for the specified application.

## Committing Changes

A commit operation saves the pending database for all application peers and releases the lock for all switches.

The commit function does not start a session; only a lock function starts a session. However, an empty commit is allowed if configuration changes are not previously made. In this case, a commit operation results in a session that acquires locks and distributes the current database.

When you commit configuration changes to a feature using the CFS infrastructure, you receive a notification about one of the following responses:

- One or more external switches report a successful status—The application applies the changes locally and releases the network lock.
- None of the external switches report a successful state—The application considers this state a failure and does not apply the changes to any switch in the network. The network lock is not released.

You can commit changes for a specified feature by entering the **commit** command for that feature.

## Discarding Changes

If you discard configuration changes, the application flushes the pending database and releases locks in the network. Both the abort and commit functions are supported only from the switch from which the network lock is acquired.

You can discard changes for a specified feature by using the **abort** command for that feature.

## Saving the Configuration

Configuration changes that have not been applied yet (still in the pending database) are not shown in the running configuration. The configuration changes in the pending database overwrite the configuration in the effective database when you commit the changes.

**Caution** If you do not commit the changes, they are not saved to the running configuration.

## Clearing a Locked Session

You can clear locks held by an application from any switch in the network to recover from situations where locks are acquired and not released. This function requires Admin permissions.

**Caution** Exercise caution when using this function to clear locks in the network. Any pending configurations in any switch in the network is flushed and lost.

# CFS Regions

## About CFS Regions

A CFS region is a user-defined subset of switches for a given feature or application in its physical distribution scope. When a network spans a vast geography, you might need to localize or restrict the distribution of certain profiles among a set of switches based on their physical proximity. CFS regions allow you to create multiple islands of distribution within the network for a given CFS feature or application. CFS regions are designed to restrict the distribution of a feature's configuration to a specific set or grouping of switches in a network.

**Note** You can only configure a CFS region based on physical switches. You cannot configure a CFS region in a VSAN.

## Example Scenario

The Smart Call Home application triggers alerts to network administrators when a situation arises or something abnormal occurs. When the network covers many geographies, and there are multiple network administrators who are each responsible for a subset of switches in the network, the Smart Call Home application sends alerts to all network administrators regardless of their location. For the Smart Call Home application to send message alerts selectively to network administrators, the physical scope of the application has to be fine tuned or narrowed down. You can achieve this scenario by implementing CFS regions.

CFS regions are identified by numbers ranging from 0 through 200. Region 0 is reserved as the default region and contains every switch in the network. You can configure regions from 1 through 200. The default region maintains backward compatibility.

If the feature is moved, that is, assigned to a new region, its scope is restricted to that region; it ignores all other regions for distribution or merging purposes. The assignment of the region to a feature has precedence in distribution over its initial physical scope.

You can configure a CFS region to distribute configurations for multiple features. However, on a given switch, you can configure only one CFS region at a time to distribute the configuration for a given feature. Once you assign a feature to a CFS region, its configuration cannot be distributed within another CFS region.

## Managing CFS Regions

### Creating CFS Regions

You can create a CFS region.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **cfs region** *region-id* | Creates a region. |

## Assigning Applications to CFS Regions

You can assign an application on a switch to a region.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure** terminal | Enters global configuration mode. |
| **Step 2** | switch(config)# **cfs region** *region-id* | Creates a region. |
| **Step 3** | switch(config-cfs-region)# *application* | Adds application(s) to the region.<br>**Note**<br>You can add any number of applications on the switch to a region. If you try adding an application to the same region more than once, you see the "Application already present in the same region" error message. |

### Example

The following example shows how to assign applications to a region:

```
switch# configure terminal
switch(config)# cfs region 1
switch(config-cfs-region)# ntp
switch(config-cfs-region)# callhome
```

## Moving an Application to a Different CFS Region

You can move an application from one region to another region.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | switch(config)# **cfs region** *region-id* | Enters CFS region configuration submode. |
| **Step 3** | switch(config-cfs-region)# *application* | Indicates application(s) to be moved from one region into another.<br>**Note**<br>If you try moving an application to the same region more than once, you see the "Application already present in the same region" error message. |

### Example

The following example shows how to move an application into Region 2 that was originally assigned to Region 1:

```
switch# configure terminal
switch(config)# cfs region 2
switch(config-cfs-region)# ntp
```

## Removing an Application from a Region

Removing an application from a region is the same as moving the application back to the default region (Region 0), which brings the entire network into the scope of distribution for the application.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **cfs region** *region-id* | Enters CFS region configuration submode. |
| **Step 3** | switch(config-cfs-region)# **no** *application* | Removes application(s) that belong to the region. |

## Deleting CFS Regions

Deleting a region nullifies the region definition. All the applications bound by the region are released back to the default region.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | switch(config)# **no cfs region** *region-id* | Deletes the region.<br>**Note**<br>You see the, "All the applications in the region will be moved to the default region" warning. |

# Configuring CFS over IP

## Enabling CFS over IPv4

You can enable or disable CFS over IPv4.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **cfs ipv4 distribute** | Globally enables CFS over IPv4 for all applications on the switch. |
| **Step 3** | (Optional) switch(config)# **no cfs ipv4 distribute** | Disables (default) CFS over IPv4 on the switch. |

## Verifying the CFS Over IP Configuration

The following example show how to verify the CFS over IP configuration:

```
switch# show cfs status
Distribution : Enabled
Distribution over IP : Enabled - mode IPv4
IPv4 multicast address : 239.255.70.83
```

## Configuring IP Multicast Addresses for CFS over IP

All CFS over IP enabled switches with similar multicast addresses form one CFS over IP network. CFS protocol-specific distributions, such as the keepalive mechanism for detecting network topology changes, use the IP multicast address to send and receive information.

**Note** CFS distributions for application data use directed unicast.

## Configuring IPv4 Multicast Address for CFS

You can configure a CFS over IP multicast address value for IPv4. The default IPv4 multicast address is 239.255.70.83.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **cfs ipv4 mcast-address** *ipv4-address* | Configures the IPv4 multicast address for CFS distribution over IPv4. The ranges of valid IPv4 addresses are 239.255.0.0 through 239.255.255.255 and 239.192/16 through 239.251/16. |
| **Step 3** | (Optional) switch(config)# **no cfs ipv4 mcast-address** *ipv4-address* | Reverts to the default IPv4 multicast address for CFS distribution over IPv4. The default IPv4 multicast address for CFS is 239.255.70.83. |

# Verifying the IP Multicast Address Configuration for CFS over IP

The following example shows how to verify the IP multicast address configuration for CFS over IP:

```
switch# show cfs status
Fabric distribution Enabled
IP distribution Enabled mode ipv4
IPv4 multicast address : 10.1.10.100
```

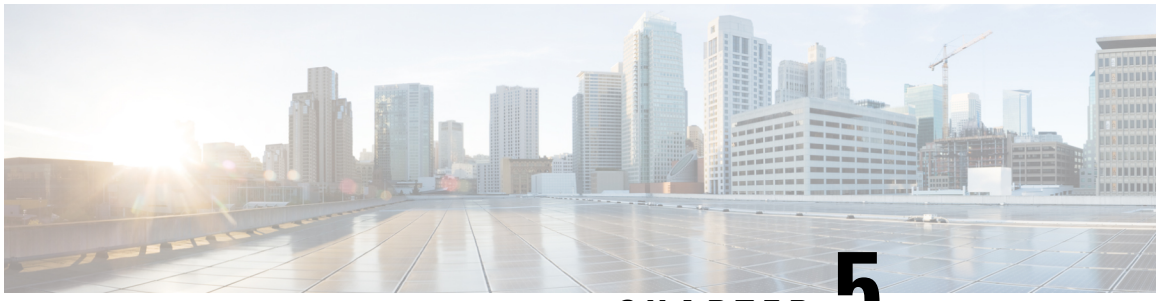
# Default Settings for CFS

The following table lists the default settings for CFS configurations.

*Table 1: Default CFS Parameters*

| Parameters | Default |
|---|---|
| CFS distribution on the switch | Enabled |
| Database changes | Implicitly enabled with the first configuration change |
| Application distribution | Differs based on application |
| Commit | Explicit configuration is required |
| CFS over IP | Disabled |
| IPv4 multicast address | 239.255.70.83 |

The CISCO-CFS-MIB contains SNMP configuration information for any CFS-related functions. See the MIB reference for your platform.



CHAPTER 5

# Configuring PTP

This chapter contains the following sections:


- Information About PTP, on page 41
- PTP Device Types, on page 41
- PTP Process, on page 42
- High Availability for PTP, on page 43
- Guidelines and Limitations for PTP, on page 43
- Default Settings for PTP, on page 43
- Configuring PTP, on page 44


## Information About PTP

PTP is a time synchronization protocol for nodes distributed across a network. Its hardware timestamp feature provides greater accuracy than other time synchronization protocols such as the Network Time Protocol (NTP).

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.

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.

PTP is not supported on Cisco Nexus 3100 switches from release 6.0(2)U3(1) through release 7.0(3)I2(4). However PTP is supported on Cisco Nexus 3100 switches from release 7.0(3)I4(1) and higher.

## PTP Device Types

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 that are 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. We recommend that you deploy a Grand Master Clock (10 MHz) upstream. The servers contain clocks that require synchronization and are connected to the switch.

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

# PTP 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

After the master-slave hierarchy has been established, the clocks are synchronized as follows:

- The master sends a synchronization message to the slave and notes the time it was sent.
- The slave receives the synchronization message and notes the time that it was received. For every synchronization message, there is a follow-up message. The number of sync messages should be equal to the number of follow-up messages.

- The slave sends a delay-request message to the master and notes the time it was sent.
- The master receives the delay-request message and notes the time it was received.
- The master sends a delay-response message to the slave. The number of delay request messages should be equal to the number of delay response messages.
- The slave uses these timestamps to adjust its clock to the time of its master.

# High Availability for PTP

Stateful restarts are not supported for PTP.

# Guidelines and Limitations for PTP

- For Cisco Nexus 3000 and 3100 Series switches, PTP clock correction is expected to be in the 3-digit range, from 100 to 999 nanoseconds.
- PTP operates only in boundary clock mode. End-to-end transparent clock and peer-to-peer transparent clock modes are not supported.
- PTP supports transport over User Datagram Protocol (UDP). Transport over Ethernet is not supported.
- PTP supports only multicast communication. Negotiated unicast communication is not supported.
- PTP is limited to a single domain per network.
- Forwarding PTP management packets is not supported.
- PTP-capable ports do not identify PTP packets and do not time-stamp or redirect those packets unless you enable PTP on those ports.
- 1 packet per second (1 pps) input is not supported.
- PTP over IPv6 is not supported.
- Cisco Nexus switches should be synchronized from the neighboring master using a synchronization log interval that ranges from –2 to –5.
- One-step PTP is not supported on Cisco Nexus 3000 and 3500 series platform switches.

# Default Settings for PTP

The following table lists the default settings for PTP parameters.

***Table 2: Default PTP Parameters***

| Parameters | Default |
|---|---|
| PTP | Disabled |
| PTP version | 2 |

| Parameters | Default |
|---|---|
| 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 sync interval | – 2 log seconds |
| PTP announce timeout | 3 announce intervals |
| PTP minimum delay request interval | 0 log seconds |
| PTP VLAN | 1 |

# Configuring PTP

## Configuring 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.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # [**no**] **feature ptp** | Enables or disables PTP on the device.<br>**Note**<br>Enabling PTP on the switch does not enable PTP on each interface. |
| **Step 3** | switch(config) # [**no**] **ptp source** *ip-address* [**vrf** *vrf*] | Configures the source IP address for all PTP packets.<br>The *ip-address* can be in IPv4 format. |
| **Step 4** | (Optional) switch(config) # [**no**] **ptp domain** *number* | Configures the domain number to use for this clock. PTP domains allow you to use multiple independent PTP clocking subdomains on a single network.<br>The range for the *number* is from 0 to 128. |
| **Step 5** | (Optional) switch(config) # [**no**] **ptp priority1** *value* | Configures the priority1 value to use when advertising this clock. This value overrides the default criteria (clock quality, clock class, and |

| | Command or Action | Purpose |
|---|---|---|
| | | so on) for the best master clock selection. Lower values take precedence.<br>The range for the *value* is from 0 to 255. |
| **Step 6** | (Optional) switch(config) # [**no**] **ptp priority2** *value* | 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.<br>The range for the *value* is from 0 to 255. |
| **Step 7** | (Optional) switch(config) # **show ptp brief** | Displays the PTP status. |
| **Step 8** | (Optional) switch(config) # **show ptp clock** | Displays the properties of the local clock. |
| **Step 9** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following 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:Sun Jul 3 14:13:24 2011
switch(config)#
```

# Configuring 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** | 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) # [**no**] **ptp** | Enables or disables PTP on an interface. |
| **Step 4** | (Optional) switch(config-if) # [**no**] **ptp announce** {**interval** *log seconds* \| **timeout** *count*} | Configures the interval between PTP announce messages on an interface or the number of PTP intervals before a timeout occurs on an interface.<br>The range for the PTP announcement interval is from 0 to 4 seconds, and the range for the interval timeout is from 2 to 10. |
| **Step 5** | (Optional) switch(config-if) # [**no**] **ptp delay request minimum interval** *log seconds* | Configures the minimum interval allowed between PTP delay-request messages when the port is in the slave state.<br>The range is from log -1 to 6 seconds. Where, log (-2) = 4 frames per second. |
| **Step 6** | (Optional) switch(config-if) # [**no**] **ptp sync interval** *log seconds* | Configures the interval between PTP synchronization messages on an interface.<br>The range for the PTP synchronization interval for Cisco Nexus 3000 Series switch is from -6 log second to 1 second.<br>The range for the PTP synchronization interval for Cisco Nexus 3548 Series switch is -3 log second to 1 second. |
| **Step 7** | (Optional) switch(config-if) # [**no**] **ptp vlan** *vlan-id* | Specifies the VLAN for the interface where PTP is being enabled. You can only enable PTP on one VLAN on an interface.<br>The range is from 1 to 4094. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 8** | (Optional) switch(config-if) # **show ptp brief** | Displays the PTP status. |
| **Step 9** | (Optional) switch(config-if) # **show ptp port interface** *interface slot/port* | Displays the status of the PTP port. |
| **Step 10** | (Optional) switch(config-if)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

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 2/1
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 4
switch(config-if)# ptp sync interval -1
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
switch(config-if)#
```

# Configuring Multiple PTP Domains

You can configure multiple PTP clocking domains on a single network. Each domain has a priority value associated with it. The default value is 255.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # [**no**] **feature ptp** | Enables or disables PTP on the device. |

| | Command or Action | Purpose |
|---|---|---|
| | | **Note** Enabling PTP on the switch does not enable PTP on each interface. |
| **Step 3** | switch(config) # [**no**] **ptp source** *ip-address* [**vrf** *vrf*] | Configures the source IP address for all PTP packets.<br>The *ip-address* can be in IPv4 format. |
| **Step 4** | switch(config) # [**no**] **ptp multi-domain** | Enables configuring multi domain feature on the switch. It also allow you to set the attributes such as priority, clock-class threshold , clock-accuracy threshold, transition priorities etc. on the switch. |
| **Step 5** | switch(config) # [**no**] **ptp domain** *value* **priority** *value* | Specify the values for the domain and priority.<br>The range for the domain *value* is from 0 to 127. The default value of the domain is 0<br>The range for the priority *value* is from 0 to 255. The default value of the priority is 255 |
| **Step 6** | switch(config) # [**no**] **ptp domain** *value* **clock-class-threshold** *value* | Specify the values for domain and clock class threshold. The default value is 248.<br>The range for the domain *value* is from 0 to 127.<br>The range for the clock-class-threshold *value* is from 0 to 255.<br>**Note** It is not necessary that a clock class threshold value ensure election of the slave clock on any ports. The switch uses this value to determine whether the source clock is traceable. If the clock class value from the peer is higher or equal than the *clock class threshold* value in a domain, the switch runs BMCA to elect the slave port from a domain. If none of the domains has the clock class below the threshold value, the switch runs BMCA on all the PTP enabled ports to elect the best clock. |
| **Step 7** | switch(config) # [**no**] **ptp domain** *value* **clock-accuracy-threshold** *value* | Specify the values for domain and clock accuracy threshold. The default value is 254.<br>The range for the domain *value* is from 0 to 127.<br>The range for the clock-accuracy-threshold *value* is from 0 to 255. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 8** | switch(config) # [**no**] **ptp multi-domain transition-attributes priority1** *value* | Sets the *domain transition-attributes priority1* value that is used when sending a packet out from this domain to a peer domain. The value of the *priority1* in the announce message from the remote port is replaced by the value of *domain transition-attributes priority1* when the announce message has to be transmitted to a peer in a domain, that is different from that of the slave interface. The default value is 255.<br>The range for the transition-attributes priority1 *value* is from 0 to 255. |
| **Step 9** | switch(config) # [**no**] **ptp multi-domain transition-attributes priority2** *value* | Sets the *domain transition-attributes priority2* value that is used when sending a packet out from this domain to a peer domain. The value of the *priority2* in the announce message from the remote port is replaced by the value of *domain transition-attributes priority2* when the announce message has to be transmitted to a peer in a domain, that is different from that of the slave interface. The default value is 255.<br>The range for the transition-attributes priority2 *value* is from 0 to 255. |
| **Step 10** | switch(config-if) # [**no**] **ptp domain** *value* | Associates a domain on a PTP enabled interface. If you do not configure the domain specifically on an interface, it takes the default value (0).<br>The range for the domain *value* is from 0 to 127. |

### Example

The following example shows the PTP domains configured on a switch:

```
switch(config)# show ptp domain data
MULTI DOMAIN : ENABLED
GM CAPABILITY : ENABLED
PTP DEFAULT DOMAIN : 0
PTP TRANSITION PRIORITY1 : 20
PTP TRANSITION PRIORITY2 : 255
PTP DOMAIN PROPERTY
Domain-Number Domain-Priority Clock-Class Clock-Accuracy Ports
0         255        248             254             Eth1/1
1           1         1              254

switch(config)#
```

The following example shows the domains associated with each PTP enabled interfaces:

```
switch(config)# show ptp interface domain
PTP port interface domain
-------------------------
Port         Domain
-------  -----------------
Eth1/1        0
      1         1              254

switch(config)#
```

# Configuring clock Identity

You can configure clock identity on a Cisco Nexus 3500 switch. The default clock identity is a unique 8-octet array presented in the form of a character array based on the switch MAC address.

.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # [**no**] **feature ptp** | Enables or disables PTP on the device.<br>**Note**<br>Enabling PTP on the switch does not enable PTP on each interface. |
| **Step 3** | switch(config-if) # **ptp clock-identity** *MAC Address* | Assigns 6 byte MAC address for PTP clock-identity. Default clock identity is based on the MAC address of the switch. The clock-identity is defined as per IEEE standard (MAC-48 Byte0 \| MAC-48 Byte1 \| MAC-48 Byte2 \| FF \| FE \| MAC-48 Bytes3-5). |

# Configuring PTP Cost Interface

You can configure interface cost on each PTP enabled port on a Cisco Nexus 3500 switch. The cost applies to each PTP enabled port if the switch has more than one path to grandmaster clock.

.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # [**no**] **feature ptp** | Enables or disables PTP on the device.<br>**Note** |

| | Command or Action | Purpose |
|---|---|---|
| | | Enabling PTP on the switch does not enable PTP on each interface. |
| **Step 3** | switch(config) # [**no**] **ptp source** *ip-address* [**vrf** *vrf*] | Configures the source IP address for all PTP packets.<br>The *ip-address* can be in IPv4 format. |
| **Step 4** | switch(config-if) # [**no**] **feature ptp** | Enables or disables PTP on the interface. |
| **Step 5** | switch(config-if) # [**no**] **ptp cost** *value* | Associate cost on a PTP enabled interface. The interface having the least cost becomes the slave interface.<br>The range for the cost is from 0 to 255. The default value is 255. |

### Example

The following example shows cost that is associated with each PTP enabled interfaces:

```
switch(config)# show ptp cost
PTP port costs
-----------------------
Port         Cost
-------  --------------
Eth1/1        255
switch(config)#
```

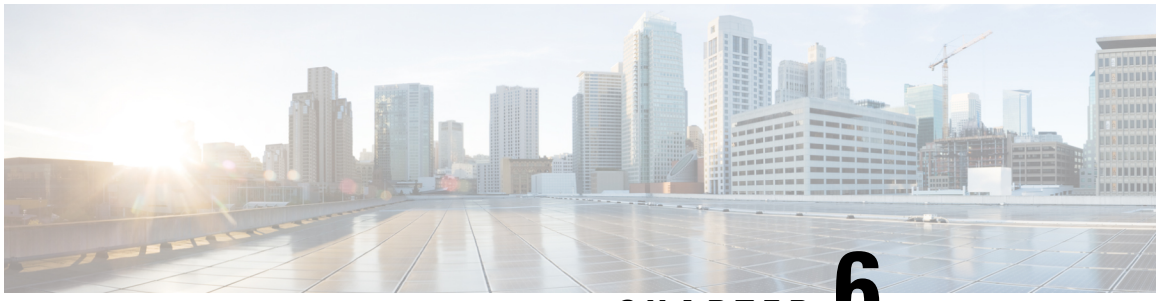
# Verifying the PTP Configuration

Use one of the following commands to verify the configuration:

***Table 3: PTP Show Commands***

| Command | Purpose |
|---|---|
| **show ptp brief** | Displays the PTP status. |
| **show ptp clock** | Displays the properties of the local clock, including the 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 parent** | Displays the properties of the PTP parent. |

| Command | Purpose |
|---|---|
| **show ptp port interface ethernet** *slot/port* | Displays the status of the PTP port on the switch. |
| **show ptp counters** [**all** \| **interface ethernet** *slot/port*] | Displays the PTP packet counters for all interfaces or for a specified interface. |
| **show ptp time-property** | Displays the PTP clock properties. |



CHAPTER 6

# Configuring NTP

This chapter contains the following sections:


- Information About NTP, on page 53
- NTP as Time Server, on page 54
- Distributing NTP Using CFS, on page 54
- Clock Manager, on page 54
- High Availability, on page 54
- Virtualization Support, on page 54
- Prerequisites for NTP, on page 55
- Guidelines and Limitations for NTP, on page 55
- Default Settings, on page 56
- Configuring NTP, on page 56
- Verifying the NTP Configuration, on page 69
- Configuration Examples for NTP, on page 69


## Information 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 as Time Server

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.

# Distributing NTP Using CFS

Cisco Fabric Services (CFS) distributes the local NTP configuration to all Cisco devices in the network.

After enabling CFS on your device, a network-wide lock is applied to NTP whenever an NTP configuration is started. After making the NTP configuration changes, you can discard or commit them.

In either case, the CFS lock is then released from the NTP application.

# Clock Manager

Clocks are resources that need to be shared across different processes.

Multiple time synchronization protocols, such as NTP and Precision Time Protocol (PTP), might be running in the system.

# High Availability

Stateless restarts are supported for NTP. After a reboot or a supervisor switchover, the running configuration is applied.

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.

# 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:

- Starting with Release 7.0(3)I2(1), the **show ntp session status** CLI command does not show the last action time stamp, the last action, the last action result, and the last action failure reason.
- NTP server functionality is supported.
- You should have 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 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, you should configure all the devices as clients to that server.
- You can configure up to 64 NTP entities (servers and peers).
- If CFS is disabled for NTP, NTP does not distribute any configuration and does not accept a distribution from other devices in the network.
- After CFS distribution is enabled for NTP, the entry of an NTP configuration command locks the network for NTP configuration until a **commit** command is entered. During the lock, no changes can be made to the NTP configuration by any other device in the network except the device that initiated the lock.
- If you use CFS to distribute NTP, all devices in the network should have the same VRFs configured as you use for NTP.
- If you configure NTP in a VRF, ensure that the NTP server and peers can reach each other through the configured VRFs.
- You must manually distribute NTP authentication keys on the NTP server and Cisco NX-OS devices across the network.
- Use NTP broadcast or multicast associations when time accuracy and reliability requirements are modest, your network is localized, and the network has more than 20 clients. We recommend that you use NTP broadcast or multicast associations in networks that have limited bandwidth, system memory, or CPU resources.
- Beginning with Cisco NX-OS Release 7.0(3)I6(1), a maximum of four ACLs can be configured for a single NTP access group.

**Note** Time accuracy is marginally reduced in NTP broadcast associations because information flows only one way.

# Default Settings

The following are the default settings for NTP parameters.

| Parameters | Default |
|---|---|
| NTP | Enabled for all interfaces |
| NTP passive (enabling NTP to form associations) | Enabled |
| NTP authentication | Disabled |
| NTP access | Enabled |
| NTP access group match all | Disabled |
| NTP broadcast server | Disabled |
| NTP multicast server | Disabled |
| NTP multicast client | Disabled |
| NTP logging | Disabled |

# Configuring NTP

## Enabling or Disabling NTP on an Interface

You can enable or disable NTP on a particular interface. NTP is enabled on all interfaces by default.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **interface** *type slot/port* | Enters interface configuration mode. |
| **Step 3** | switch(config-if)# **[no] ntp disable** {**ip** \| **ipv6**} | Disables NTP IPv4 or IPv6 on the specified interface.<br>Use the **no** form of this command to reenable NTP on the interface. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | (Optional) switch(config-if)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to enable or disable NTP on an interface:

```
switch# configure terminal
switch(config)# interface ethernet 6/1
switch(config-if)# ntp disable ip
switch(config-if)# copy running-config startup-config
```

# 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** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **ntp master** [*stratum*] | Configures the device as an authoritative NTP server.<br>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** | Displays the NTP configuration. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure the Cisco NX-OS device as an authoritative NTP server with a different stratum level:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp master 5
```

# Configuring an NTP Server and Peer

You can configure an NTP server and peer.

### Before you begin

Make sure that you know the IP address or DNS names of your NTP server and its peers.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# [**no**] **ntp server** {*ip-address* \| *ipv6-address* \| *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*] | Forms an association with a server.<br>Use the **key** keyword to configure a key to be used while communicating with the NTP server.<br>The range for the *key-id* argument is from 1 to 65535.<br>Use the **maxpoll** and **minpoll** keywords to configure the maximum and minimum intervals in which to poll a server. The range for the *max-poll* and *min-poll* 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 (*maxpoll* default = 64 seconds, *minpoll* default = 16 seconds).<br>Use the **prefer keyword** to make this the preferred NTP server for the device.<br>Use the **use-vrf** keyword to configure the NTP server to communicate over the specified VRF.<br>The *vrf-name* argument can be default, management, or any case-sensitive alphanumeric string up to 32 characters.<br>**Note**<br>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** | switch(config)# [**no**] **ntp peer** {*ip-address* \| *ipv6-address* \| *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*] | Forms an association with a peer. You can specify multiple peer associations.<br>Use the **key** keyword to configure a key to be used while communicating with the NTP peer. The range for the *key-id* argument is from 1 to 65535. |

| | Command or Action | Purpose |
|---|---|---|
| | | Use the **maxpoll** and **minpoll** keywords to configure the maximum and minimum intervals in which to poll a server. The range for the *max-poll* and *min-poll* 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 (*maxpoll* default = 64 seconds, *minpoll* default = 16 seconds).<br>Use the **prefer** keyword to make this the preferred NTP peer for the device.<br>Use the **use-vrf** keyword to configure the NTP peer to communicate over the specified VRF. The *vrf-name* argument can be **default** , **management** , or any case-sensitive alphanumeric string up to 32 characters. |
| **Step 4** | (Optional) switch(config)# **show ntp peers** | Displays the configured server and peers.<br>**Note**<br>A domain name is resolved only when you have a DNS server configured. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying 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

Authentication for NTP servers and NTP peers is configured on a per-association basis using the **key** keyword on each **ntp server** and **ntp peer** command. Make sure that you configured all NTP server and peer associations with the authentication keys that you plan to specify in this procedure. Any **ntp server** or **ntp peer**commands that do not specify the **key** keyword will continue to operate without authentication.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch# configure terminal`<br>`switch(config)#` | |
| **Step 2** | [**no**] **ntp authentication-key** *number* **md5** *md5-string*<br>**Example:**<br>`switch(config)# ntp authentication-key 42 md5 aNiceKey` | Defines the authentication keys. The device does not synchronize to a time source unless the source has one of these authentication keys and the key number is specified by the **ntp trusted-key** *number* command.<br>The range for authentication keys is from 1 to 65535. For the MD5 string, you can enter up to eight alphanumeric characters. |
| **Step 3** | **ntp server** *ip-address* **key** *key-id*<br>**Example:**<br>`switch(config)# ntp server 192.0.2.1 key 1001` | Enables authentication for the specified NTP server, forming an association with a server.<br>Use the **key** keyword to configure a key to be used while communicating with the NTP server. The range for the *key-id* argument is from 1 to 65535.<br>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**<br>**Example:**<br>`switch(config)# show ntp authentication-keys` | Displays the configured NTP authentication keys. |
| **Step 5** | [**no**] **ntp trusted-key** *number*<br>**Example:**<br>`switch(config)# ntp trusted-key 42` | 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.<br>This command provides protection against accidentally synchronizing the device to a time source that is not trusted. |
| **Step 6** | (Optional) **show ntp trusted-keys**<br>**Example:**<br>`switch(config)# show ntp trusted-keys` | Displays the configured NTP trusted keys. |
| **Step 7** | [**no**] **ntp authenticate**<br>**Example:**<br>`switch(config)# ntp authenticate` | Enables or disables authentication for ntp passive, ntp broadcast client, and ntp multicast. NTP authentication is disabled by default. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 8** | (Optional) **show ntp authentication-status**<br>**Example:**<br>`switch(config)# show ntp authentication-status` | Displays the status of NTP authentication. |
| **Step 9** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | 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.

Beginning with Cisco NX-OS Release 7.0(3)I7(3), access groups are evaluated in the following method:

- 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).

The mapping of the access group to the type of packet is as follows:

- 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 descending order:

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

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | switch(config)# [**no**] **ntp access-group match-all** \| {{**peer** \| **serve** \| **serve-only** \| **query-only** }*access-list-name*} | Creates or removes an access group to control NTP access and applies a basic IP access list.<br><br>The access group options are scanned in the following order, from least restrictive to most restrictive. However, if NTP matches a deny ACL rule in a configured peer, ACL processing stops and does not continue to the next access group option.<br><br>• 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.<br><br>• 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.<br><br>• The **serve-only** keyword enables the device to receive only time requests from servers specified in the access list.<br><br>• The **query-only** keyword enables the device to receive only NTP control queries from the servers specified in the access list.<br><br>• 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.<br><br>**Note**<br>The **match-all** keyword is available beginning with Cisco NX-OS Release 7.0(3)I6(1). |
| **Step 3** | switch(config)# **show ntp access-groups** | (Optional) Displays the NTP access group configuration. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure the device to allow it to synchronize to a peer from access group "accesslist1":

```
switch# configure terminal
switch(config)# ntp access-group peer accesslist1
switch(config)# show ntp access-groups
Access List Type
----------------------------
accesslist1 Peer
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

# 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** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **ntp source** *ip-address* | Configures the source IP address for all NTP packets. The *ip-address* can be in IPv4 or IPv6 format. |

### Example

This example shows how to configure an NTP source IP address of 192.0.2.2.

```
switch# configure terminal
switch(config)# ntp source 192.0.2.2
```

# Configuring the NTP Source Interface

You can configure NTP to use a specific interface.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **ntp source-interface** *interface* | Configures the source interface for all NTP packets. The following list contains the valid values for *interface*. |

| | Command or Action | Purpose |
|---|---|---|
| | | • ethernet<br>• loopback<br>• mgmt<br>• port-channel<br>• vlan |

### Example

This example shows how to configure the NTP source interface:

```
switch# configure terminal
switch(config)# ntp source-interface ethernet
```

# Configuring an NTP Broadcast Server

You can configure an NTP IPv4 broadcast server on an interface. The device then sends broadcast packets through that interface periodically. The client is not required to send a response.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **interface** *type slot/port* | Enters interface configuration mode. |
| **Step 3** | switch(config-if)# **[no] ntp broadcast** [**destination** *ip-address*] [**key** *key-id*] [*version number*] | Enables an NTP IPv4 broadcast server on the specified interface.<br>• **destination** *ip-address*—Configures the broadcast destination IP address.<br>• **key** *key-id*—Configures the broadcast authentication key number. The range is from 1 to 65535.<br>• *version number*—Configures the NTP version. The range is from 2 to 4. |
| **Step 4** | switch(config-if)# **exit** | Exits interface configuration mode. |
| **Step 5** | (Optional) switch(config)# **[no] ntp broadcastdelay** *delay* | Configures the estimated broadcast round-trip delay in microseconds. The range is from 1 to 999999. |
| **Step 6** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure an NTP broadcast server:

```
switch# configure terminal
switch(config)# interface ethernet 6/1
switch(config-if)# ntp broadcast destination 192.0.2.10
switch(config-if)# exit
switch(config)# ntp broadcastdelay 100
switch(config)# copy running-config startup-config
```

# Configuring an NTP Multicast Server

You can configure an NTP IPv4 or IPv6 multicast server on an interface. The device then sends multicast packets through that interface periodically.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **interface** *type slot/port* | Enters interface configuration mode. |
| **Step 3** | switch(config-if)# **[no] ntp multicast** [*ipv4-address* \| *ipv6-address*] [**key** *key-id*] [*ttl value*] [*version number*] | Enables an NTP IPv4 or IPv6 multicast server on the specified interface.<br>• *ipv4-address* or *ipv6-address*— Multicast IPv4 or IPv6 address.<br>• **key** *key-id*—Configures the broadcast authentication key number. The range is from 1 to 65535.<br>• *ttl value*—Time-to-live value of the multicast packets. The range is from 1 to 255.<br>• *version number*—NTP version. The range is from 2 to 4. |
| **Step 4** | (Optional) switch(config-if)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure an Ethernet interface to send NTP multicast packets:

```
switch# configure terminal
switch(config)# interface ethernet 2/2
switch(config-if)# ntp multicast FF02::1:FF0E:8C6C
switch(config-if)# copy running-config startup-config
```

# Configuring an NTP Multicast Client

You can configure an NTP multicast client on an interface. The device then listens to NTP multicast messages and discards any messages that come from an interface for which multicast is not configured.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **interface** *type slot/port* | Enters interface configuration mode. |
| **Step 3** | switch(config-if)# **[no] ntp multicast client** [*ipv4-address* \| *ipv6-address*] | Enables the specified interface to receive NTP multicast packets. |
| **Step 4** | (Optional) switch(config-if)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure an Ethernet interface to receive NTP multicast packets:

```
switch# configure terminal
switch(config)# interface ethernet 2/3
switch(config-if)# ntp multicast client FF02::1:FF0E:8C6C
switch(config-if)# copy running-config startup-config
```

# 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** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# [**no**] **ntp logging** | Enables or disables system logs to be generated with significant NTP events. NTP logging is disabled by default. |
| **Step 3** | (Optional) switch(config)# **show ntp logging-status** | Displays the NTP logging configuration status. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to enable NTP logging in order to generate system logs with significant NTP events:

```
switch# configure terminal
switch(config)# ntp logging
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

# Enabling CFS Distribution for NTP

You can enable CFS distribution for NTP in order to distribute the NTP configuration to other CFS-enabled devices.

### Before you begin

Make sure that you have enabled CFS distribution for the device.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# [**no**] **ntp distribute** | Enables or disables the device to receive NTP configuration updates that are distributed through CFS. |
| **Step 3** | (Optional) switch(config)# **show ntp status** | Displays the NTP CFS distribution status. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to enable the device to receive NTP configuration updates through CFS:

```
switch# configure terminal
switch(config)# ntp distribute
switch(config)# copy running-config startup-config
```

# Committing NTP Configuration Changes

When you commit the NTP configuration changes, the effective database is overwritten by the configuration changes in the pending database and all the devices in the network receive the same configuration.

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **ntp commit** | Distributes the NTP configuration changes to all Cisco NX-OS devices in the network and releases the CFS lock. This command overwrites the effective database with the changes made to the pending database. |

# Discarding NTP Configuration Changes

After making the configuration changes, you can choose to discard the changes instead of committing them. If you discard the changes, Cisco NX-OS removes the pending database changes and releases the CFS lock.

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **ntp abort** | Discards the NTP configuration changes in the pending database and releases the CFS lock. Use this command on the device where you started the NTP configuration. |

# Releasing the CFS Session Lock

If you have performed an NTP configuration and have forgotten to release the lock by either committing or discarding the changes, you or another administrator can release the lock from any device in the network. This action also discards pending database changes.

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **clear ntp session** | Discards the NTP configuration changes in the pending database and releases the CFS lock. |

# Verifying the NTP Configuration

| 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 peer** | Displays all the NTP peers. |
| **show ntp pending** | Displays the temporary CFS database for NTP. |
| **show ntp pending-diff** | Displays the difference between the pending CFS database and the current NTP configuration. |
| **show ntp rts-update** | Displays the RTS update status. |
| **show ntp session status** | Displays the NTP CFS distribution session information. |
| **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*} \| **name** *peer-name*}} | Displays the NTP statistics. |
| **show ntp status** | Displays the NTP CFS distribution status. |
| **show ntp trusted-keys** | Displays the configured NTP trusted keys. |
| **show running-config ntp** | Displays NTP information. |

# Configuration Examples for NTP

### Configuration Examples for NTP

This example shows how to configure an NTP server and peer, enable NTP authentication, enable NTP logging, and then save the startup configuration so that it is saved across reboots and restarts:

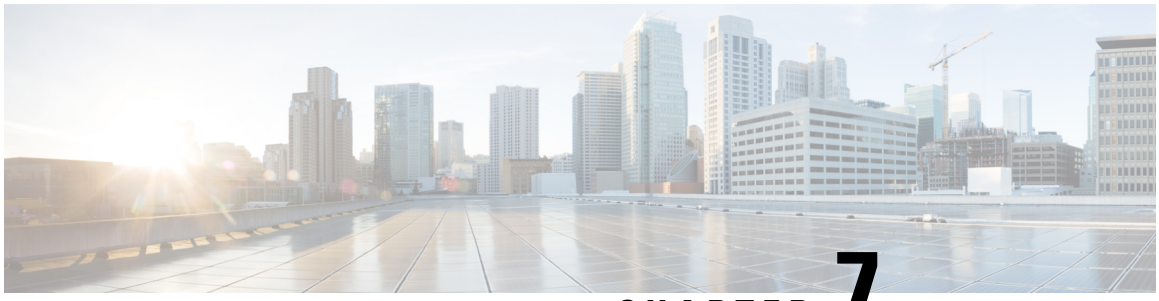
```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp server 192.0.2.105 key 42
switch(config)# ntp peer 192.0.2.105
switch(config)# show ntp peers
--------------------------------------------------
```

```
Peer IP Address Serv/Peer
-------------------------------------------------
192.0.2.100 Peer (configured)
192.0.2.105 Server (configured)
switch(config)# ntp authentication-key 42 md5 aNiceKey
switch(config)# show ntp authentication-keys
-----------------------------
Auth key MD5 String
-----------------------------
42 aNicekey
switch(config)# ntp trusted-key 42
switch(config)# show ntp trusted-keys
Trusted Keys:
42
switch(config)# ntp authenticate
switch(config)# show ntp authentication-status
Authentication enabled.
switch(config)# ntp logging
switch(config)# show ntp logging
NTP logging enabled.
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
```



CHAPTER 7

# Configuring System Message Logging

This chapter contains the following sections:


- Information About System Message Logging, on page 71
- Guidelines and Limitations for System Message Logging, on page 72
- Default Settings for System Message Logging, on page 72
- Configuring System Message Logging, on page 73
- Verifying the System Message Logging Configuration, on page 87


## Information 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 Cisco Nexus device outputs messages to terminal sessions.

By default, the switch 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 4: System Message Severity Levels***

| Level | Description |
|---|---|
| 0 – emergency | System unusable |
| 1 – alert | Immediate action needed |
| 2 – critical | Critical condition |
| 3 – error | Error condition |
| 4 – warning | Warning condition |
| 5 – notification | Normal but significant condition |

| Level | Description |
|---|---|
| 6 – informational | Informational message only |
| 7 – debugging | Appears during debugging only |

The switch 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

Syslog servers run on remote systems that are configured to log system messages based on the syslog protocol. You can configure the Cisco Nexus Series switch to sends logs to up to eight syslog servers. If CFS is enabled, you can configure up to three 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 switch first initializes, messages are sent to syslog servers only after the network is initialized.

# Guidelines and Limitations for System Message Logging

See the following guidelines and limitations for System Message Logging:

- System messages are logged to the console and to the logfile by default.
- In releases prior to Release 7.0(3)I2(1), there was no syslog message indicating the MAC collision events. Starting 7.0(3)I2(1) there is a new syslog on Cisco Nexus 3000 Series platforms to indicate the MAC collision events. The syslog message has the details, for example, the source MAC address, the VLANs, and the internal port number information. MAC collisions are normal and they are expected if the table usage crosses about 75% as observed on various setups. See the following example of the syslog: `2015 Mar 26 06:20:37 switch%-SLOT1-5-BCM_L2_HASH_COLLISION: L2 ENTRY unit=0 mac=00:11:11:f7:46:40 vlan=1998 port=0x0800082e`.

# Default Settings for System Message Logging

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

***Table 5: Default System Message Logging Parameters***

| Parameters | Default |
|---|---|
| Console logging | Enabled at severity level 2 |

| Parameters | Default |
|---|---|
| Monitor logging | Enabled at severity level 2 |
| 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

## Configuring System Message Logging to Terminal Sessions

You can configure the switch to log messages by their severity level to console, Telnet, and Secure Shell sessions.

By default, logging is enabled for terminal sessions.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **terminal monitor** | Copies syslog messages from the console to the current terminal session. |
| **Step 2** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 3** | switch(config)# **logging console** [*severity-level*] | Enables the switch to log messages to the console session based on a specified severity level or higher (a lower number value indicates a higher severity level). Severity levels range from 0 to 7:<br>• 0 – emergency<br>• 1 – alert<br>• 2 – critical<br>• 3 – error<br>• 4 – warning<br>• 5 – notification |

| | Command or Action | Purpose |
|---|---|---|
| | | • 6 – informational<br>• 7 – debugging<br>If the severity level is not specified, the default of 2 is used. |
| **Step 4** | (Optional) switch(config)# **no logging console** [*severity-level*] | Disables logging messages to the console. |
| **Step 5** | switch(config)# **logging monitor** [*severity-level*] | Enables the switch to log messages to the monitor based on a specified severity level or higher (a lower number value indicates a higher severity level). Severity levels range from 0 to 7:<br>• 0 – emergency<br>• 1 – alert<br>• 2 – critical<br>• 3 – error<br>• 4 – warning<br>• 5 – notification<br>• 6 – informational<br>• 7 – debugging<br>If the severity level is not specified, the default of 2 is used.<br>The configuration applies to Telnet and SSH sessions. |
| **Step 6** | (Optional) switch(config)# **no logging monitor** [*severity-level*] | Disables logging messages to Telnet and SSH sessions. |
| **Step 7** | (Optional) switch# **show logging console** | Displays the console logging configuration. |
| **Step 8** | (Optional) switch# **show logging monitor** | Displays the monitor logging configuration. |
| **Step 9** | (Optional) switch# **copy running-config startup-config** | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to configure a logging level of 3 for the console:

```
switch# configure terminal
switch(config)# logging console 3
```

The following example shows how to display the console logging configuration:

```
switch# show logging console
Logging console:                enabled (Severity: error)
```

The following example shows how to disable logging for the console:

```
switch# configure terminal
switch(config)# no logging console
```

The following example shows how to configure a logging level of 4 for the terminal session:

```
switch# terminal monitor
switch# configure terminal
switch(config)# logging monitor 4
```

The following example shows how to display the terminal session logging configuration:

```
switch# show logging monitor
Logging monitor:                enabled (Severity: warning)
```

The following example shows how to disable logging for the terminal session:

```
switch# configure terminal
switch(config)# no logging monitor
```

# Configuring System Message Logging to a File

You can configure the switch to log system messages to a file. By default, system messages are logged to the file log:messages.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **logging logfile** *logfile-name severity-level* [**size** *bytes*] | Configures the name of the log file used to store system messages and the minimum severity level to log. You can optionally specify a maximum file size. The default severity level is 5 and the file size is 4194304.<br>Severity levels range from 0 to 7:<br>• 0 – emergency<br>• 1 – alert<br>• 2 – critical |

| | Command or Action | Purpose |
|---|---|---|
| | | • 3 – error<br>• 4 – warning<br>• 5 – notification<br>• 6 – informational<br>• 7 – debugging<br>The file size is from 4096 to 10485760 bytes. |
| **Step 3** | (Optional) switch(config)# **no logging logfile** [*logfile-name severity-level* [**size** *bytes*]] | Disables logging to the log file. You can optionally specify a maximum file size. The default severity level is 5 and the file size is 4194304. |
| **Step 4** | (Optional) switch# **show logging info** | Displays the logging configuration. You can optionally specify a maximum file size. The default severity level is 5 and the file size is 4194304. |
| **Step 5** | (Optional) switch# **copy running-config startup-config** | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to configure a switch to log system messages to a file:

```
switch# configure terminal
switch(config)# logging logfile my_log 6 size 4194304
```

The following example shows how to display the logging configuration (some of the output has been removed for brevity):

```
switch# show logging info
Logging console:                 enabled (Severity: debugging)
Logging monitor:                 enabled (Severity: debugging)

Logging timestamp:               Seconds
Logging server:                  disabled
Logging logfile:                 enabled
        Name - my_log: Severity - informational Size - 4194304
Facility        Default Severity        Current Session Severity
--------        ----------------        ------------------------
aaa                     3                       3
aclmgr                        3                             3
afm                     3                       3
altos                   3                       3
auth                    0                       0
authpriv                3                       3
bootvar                 5                       5
callhome                2                       2
capability              2                       2
cdp                     2                       2
```

```
cert_enroll             2                       2
...
```

# 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** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **logging module** [*severity-level*] | Enables module log messages that have the specified severity level or higher. Severity levels range from 0 to 7:<br>• 0 – emergency<br>• 1 – alert<br>• 2 – critical<br>• 3 – error<br>• 4 – warning<br>• 5 – notification<br>• 6 – informational<br>• 7 – debugging<br>If the severity level is not specified, the default of 5 is used. |
| **Step 3** | switch(config)# **logging level** *facility severity-level* | Enables logging messages from the specified facility that have the specified severity level or higher. Severity levels from 0 to 7:<br>• 0 – emergency<br>• 1 – alert<br>• 2 – critical<br>• 3 – error<br>• 4 – warning<br>• 5 – notification<br>• 6 – informational<br>• 7 – debugging |

| | Command or Action | Purpose |
|---|---|---|
| | | To apply the same severity level to all facilities, use the all facility. For defaults, see the **show logging level** command.<br>**Note** Starting with Release 7.0(3)I2(1), you cannot configure the logging level for the BCM_USD, ETHPC, FWM, and NOHMS processes. For the BCM_USD process, use **attach module 1** command and then configure the logging level.<br>**Note** If the default severity and the current session severity of a component is same, then it is expected to not see the logging level for the component in the running configuration. The default logging level is not displayed in the running configuration, but it is displayed in the **show logging level** command. |
| **Step 4** | (Optional) switch(config)# **no logging module** [*severity-level*] | Disables module log messages. |
| **Step 5** | (Optional) switch(config)# **no logging level** [*facility severity-level*] | Resets the logging severity level for the specified facility to its default level. If you do not specify a facility and severity level, the switch resets all facilities to their default levels. |
| **Step 6** | (Optional) switch# **show logging module** | Displays the module logging configuration. |
| **Step 7** | (Optional) switch# **show logging level** [*facility*] | Displays the logging level configuration and the system default level by facility. If you do not specify a facility, the switch displays levels for all facilities. |
| **Step 8** | (Optional) switch# **copy running-config startup-config** | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to configure the severity level of module and specific facility messages:

```
switch# configure terminal
switch(config)# logging module 3
switch(config)# logging level aaa 2
```

# Configuring Logging Timestamps

You can configure the time-stamp units of messages logged by the Cisco Nexus Series switch.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **logging timestamp** {**microseconds** \| **milliseconds** \| **seconds**} | Sets the logging time-stamp units. By default, the units are seconds. |
| **Step 3** | (Optional) switch(config)# **no logging timestamp** {**microseconds** \| **milliseconds** \| **seconds**} | Resets the logging time-stamp units to the default of seconds. |
| **Step 4** | (Optional) switch# **show logging timestamp** | Displays the logging time-stamp units configured. |
| **Step 5** | (Optional) switch# **copy running-config startup-config** | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to configure the time-stamp units of messages:

```
switch# configure terminal
switch(config)# logging timestamp milliseconds
switch(config)# exit
switch# show logging timestamp
Logging timestamp:              Milliseconds
```

# Configuring the ACL Logging Cache

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **logging ip access-list cache entries** *num_entries* | Sets the maximum number of log entries cached in software. The range is from 0 to 1000000 entries. The default value is 8000 entries. |
| **Step 3** | switch(config)# **logging ip access-list cache interval** *seconds* | Sets the number of seconds between log updates. Also if an entry is inactive for this duration, it is removed from the cache. The range is from 5 to 86400 seconds. The default value is 300 seconds. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | switch(config)# **logging ip access-list cache threshold** *num_packets* | Sets the number of packet matches before an entry is logged. The range is from 0 to 1000000 packets. The default value is 0 packets, which means that logging is not triggered by the number of packet matches. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example show how to set the maximum number of log entries to 5000, the interval to 120 seconds, and the threshold to 500000:

```
switch# configure terminal
switch(config)# logging ip access-list cache entries 5000
switch(config)# logging ip access-list cache interval 120
switch(config)# logging ip access-list cache threshold 500000
switch(config)# copy running-config startup-config
```

# Applying ACL Logging to an Interface

### Before you begin

- Create an IP access list with at least one access control entry (ACE) configured for logging.
- Configure the ACL logging cache.
- Configure the ACL log match level.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **interface mgmt0** | Specifies the mgmt0 interface. |
| **Step 3** | switch(config-if)# **ip access-group** *name* **in** | Enables ACL logging on ingress traffic for the specified interface. |
| **Step 4** | (Optional) switch(config-if)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to apply the mgmt0 interface with the logging specified in acl1 for all ingress traffic:

```
switch# configure terminal
switch(config)# interface mgmt0
switch(config-if)# ip access-group acl1 in
switch(config-if)# copy running-config startup-config
```

# Configuring a Logging Source-Interface

You can set all system logging (syslog) messages that are sent to syslog servers to contain the same IP address as the source address, regardless of which interface the syslog message uses to exit the router. The system allows a user-configured source-IP in a syslog packet specified by the source-interface.

**Note** If a valid IP address is not assigned, the syslog is thrown and messages are sent out carrying the exit interfaces IP address.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# [**no**] **logging source-interface** [**ethernet** *slot*/*port* \| **loopback** *interface-number* \| **mgmt** *interface-number* \| **port-channel** *port channel-number* \| **vlan** *interface-number* \| **tunnel** *interface-number*] | • **ethernet**—The range for the Ethernet option source-interface is from 1 to 253.<br>• **loopback**—The range for the loopback option source-interface is from 1 to 1023.<br>• **mgmt**—The interface number for the management option source-interface is 0.<br>• **port-channel**—The range for the port channel option source-interface is from 1 to 4096. |
| **Step 3** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure the source-interface as the ethernet interface:

```
switch# configure terminal
switch(config)# logging source-interface ethernet 2/1
switch(config)# copy running-config startup-config
```

# Configuring the ACL Log Match Level

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **acllog match-log-level** *number* | Specifies the logging level to match for entries to be logged in the ACL log (acllog). The *number* is a value from 0 to 7. The default is 6.<br>**Note**<br>For log messages to be entered in the logs, the logging level for the ACL log facility (acllog) and the logging severity level for the logfile must be greater than or equal to the ACL log match log level setting. For more information, see Configuring Module and Facility Messages Logging, on page 77 and Configuring System Message Logging to a File, on page 75. |
| **Step 3** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

# Configuring Syslog Servers

You can configure up to eight syslog servers that reference remote systems where you want to log system messages.

**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 3000 Series NX-OS Unicast Routing Configuration Guide.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **logging server** *host* [*severity-level* [**use-vrf** *vrf-name* [**facility** *facility*]]] | Configures a host to receive syslog messages. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`switch(config)# logging server 172.28.254.254 5 use-vrf default facility local3` | • The *host* argument identifies the hostname or the IPv4 or IPv6 address of the syslog server host.<br>• The *severity-level* argument limits the logging of messages to the syslog server to a specified level. Severity levels range from 0 to 7. See Table 4: System Message Severity Levels , on page 71.<br>• The **use vrf** *vrf-name* keyword identifies the default or management values for the VRF name. If a specific VRF is not identified, management is the default.<br>The **show running** command output can display or not display the VRF based on the following configuration scenarios:<br>• You have not configured any VRF and the system takes the management VRF as the default. Then this VRF is not displayed in the output.<br>• You have configured management VRF. Then this VRF is not displayed in the output as the system identifies it as the default.<br>• You have configured any other VRF. Then this VRF is displayed in the output.<br>**Note**<br>The current Cisco Fabric Services (CFS) distribution does not support VRF. If CFS distribution is enabled, the logging server configured with the default VRF is distributed as the management VRF.<br>• The facility argument names the syslog facility type. The default outgoing facility is local7.<br>The facilities are listed in the command reference for the Cisco Nexus Series software that you are using.<br>**Note**<br>Debugging is a CLI facility but the debug syslogs are not sent to the server. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | (Optional) **no logging server** *host*<br>**Example:**<br>`switch(config)# no logging server 172.28.254.254 5` | Removes the logging server for the specified host. |
| **Step 4** | (Optional) **show logging server**<br>**Example:**<br>`switch# show logging server` | Displays the syslog server configuration. |
| **Step 5** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following examples show how to configure a syslog server:

```
switch# configure terminal
switch(config)# logging server 172.28.254.254 5
use-vrf default facility local3

switch# configure terminal
switch(config)# logging server 172.28.254.254 5 use-vrf management facility local3
```

## Configuring syslog 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 6: 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.<br>**Note**<br>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. |

| Field | Description |
|---|---|
| Action | Destination for messages, which can be a filename, a hostname preceded by the at sign (@), or 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:

```
debug.local7             /var/log/myfile.log
```

**Step 2** Create the log file by entering these commands at the shell prompt:

```
$ touch /var/log/myfile.log
$ chmod 666 /var/log/myfile.log
```

**Step 3** Make sure that the system message logging daemon reads the new changes by checking myfile.log after entering this command:

```
$ kill -HUP ~cat /etc/syslog.pid~
```

# Configuring syslog Server Configuration Distribution

You can distribute the syslog server configuration to other switches in the network by using the Cisco Fabric Services (CFS) infrastructure.

After you enable syslog server configuration distribution, you can modify the syslog server configuration and view the pending changes before committing the configuration for distribution. As long as distribution is enabled, the switch maintains pending changes to the syslog server configuration.

**Note** If the switch is restarted, the syslog server configuration changes that are kept in volatile memory might get lost.

### Before you begin

You must have configured one or more syslog servers.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | switch(config)# **logging distribute** | Enables distribution of the syslog server configuration to network switches using the CFS infrastructure. By default, distribution is disabled. |
| **Step 3** | switch(config)# **logging commit** | Commits the pending changes to the syslog server configuration for distribution to the switches in the fabric. |
| **Step 4** | switch(config)# **logging abort** | Cancels the pending changes to the syslog server configuration. |
| **Step 5** | (Optional) switch(config)# **no logging distribute** | Disables the distribution of the syslog server configuration to network switches using the CFS infrastructure. You cannot disable distribution when configuration changes are pending. See the **logging commit** and **logging abort** commands. By default, distribution is disabled. |
| **Step 6** | (Optional) switch# **show logging pending** | Displays the pending changes to the syslog server configuration. |
| **Step 7** | (Optional) switch# **show logging pending-diff** | Displays the differences from the current syslog server configuration to the pending changes of the syslog server configuration. |
| **Step 8** | (Optional) switch# **copy running-config startup-config** | Copies the running configuration to the startup configuration. |

# 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** | switch# **show logging last** *number-lines* | 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** | switch# **show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*] | Displays the messages in the log file that have a time stamp 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 3** | switch# **show logging nvram** [**last** *number-lines*] | Displays the messages in the NVRAM. To limit the number of lines displayed, you can enter the |

| | Command or Action | Purpose |
|---|---|---|
| | | last number of lines to display. You can specify from 1 to 100 for the last number of lines. |
| **Step 4** | switch# **clear logging logfile** | Clears the contents of the log file. |
| **Step 5** | switch# **clear logging nvram** | Clears the logged messages in NVRAM. |

### Example

The following example shows how to display messages in a log file:

```
switch# show logging last 40
switch# show logging logfile start-time 2007 nov 1 15:10:0
switch# show logging nvram last 10
```

The following example shows how to clear messages in a log file:

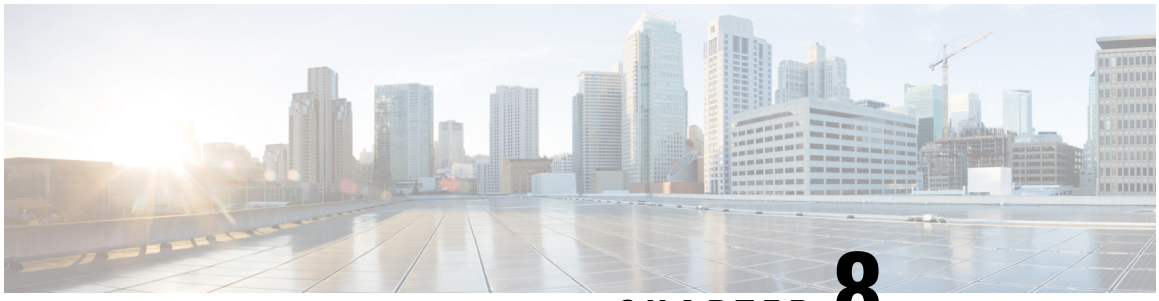
```
switch# clear logging logfile
switch# clear logging nvram
```

# Verifying the System Message Logging Configuration

Use these commands to verify system message logging configuration information:

| Command | Purpose |
|---|---|
| **show logging console** | Displays the console logging configuration. |
| **show logging info** | Displays the logging configuration. |
| **show logging ip access-list cache** | Displays the IP access list cache. |
| **show logging ip access-list cache detail** | Displays detailed information about the IP access list cache. |
| **show logging ip access-list status** | Displays the status of the IP access list cache. |
| **show logging last** *number-lines* | Displays the last number of lines of the log file. |
| **show logging level** [*facility*] | Displays the facility logging severity level configuration. |
| **show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*] | Displays the messages in the log file. |
| **show logging module** | Displays the module logging configuration. |
| **show logging monitor** | Displays the monitor logging configuration. |
| **show logging nvram** [**last** *number-lines*] | Displays the messages in the NVRAM log. |

| Command | Purpose |
|---|---|
| **show logging pending** | Displays the syslog server pending distribution configuration. |
| **show logging pending-diff** | Displays the syslog server pending distribution configuration differences. |
| **show logging server** | Displays the syslog server configuration. |
| **show logging session** | Displays the logging session status. |
| **show logging status** | Displays the logging status. |
| **show logging timestamp** | Displays the logging time-stamp units configuration. |
| **show running-config acllog** | Displays the running configuration for the ACL log file. |



CHAPTER 8

# Configuring Smart Call Home

This chapter contains the following sections:


- Information About Smart Call Home, on page 89
- Guidelines and Limitations for Smart Call Home, on page 97
- Prerequisites for Smart Call Home, on page 97
- Default Call Home Settings, on page 98
- Configuring Smart Call Home, on page 98
- Verifying the Smart Call Home Configuration, on page 108
- Sample Syslog Alert Notification in Full-Text Format, on page 109
- Sample Syslog Alert Notification in XML Format, on page 109


## Information About Smart Call Home

Smart Call Home provides e-mail-based notification of critical system events. Cisco Nexus Series switches provide a range of message formats for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications. You can use this feature to page a network support engineer, e-mail a Network Operations Center, or use Cisco Smart Call Home services to automatically generate a case with the Technical Assistance Center (TAC).

If you have a service contract directly with Cisco, you can register your devices for the Smart Call Home service. Smart Call Home provides fast resolution of system problems by analyzing Smart Call Home messages sent from your devices and providing background information and recommendations. For issues that can be identified as known, particularly GOLD diagnostics failures, Automatic Service Requests will be generated by 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 from your device and, where appropriate, Automatic Service Request generation, routed to the appropriate TAC team, including detailed diagnostic information to speed problem resolution.
- Secure message transport directly from your device or through a downloadable Transport Gateway (TG) aggregation point. You can use a TG aggregation point in cases that require support for multiple devices or in cases where security requirements mandate 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, and field notices, security advisories, and end-of-life information.

# Smart Call Home Overview

You can use Smart Call Home to notify an external entity when an important event occurs on your device. Smart Call Home delivers alerts to multiple recipients that you configure in destination profiles.

Smart Call Home includes a fixed set of predefined alerts on your switch. These alerts are grouped into alert groups and CLI commands that are assigned to execute when an alert in an alert group occurs. The switch includes the command output in the transmitted Smart Call Home message.

The Smart Call Home feature offers the following:

- Automatic execution and attachment of relevant CLI command output.
- Multiple message format options such as the following:
  - Short Text—Text that is suitable for pagers or printed reports.
  - Full Text—Fully formatted message information that is suitable for human reading.
  - XML—Matching readable format that uses the Extensible Markup Language (XML) and the Adaptive Messaging Language (AML) XML schema definition (XSD). The XML format enables communication with the Cisco TAC.
- Multiple concurrent message destinations. You can configure up to 50 e-mail destination addresses for each destination profile.

# Smart Call Home Destination Profiles

A Smart Call Home 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 e-mail destinations—The list of recipients for the Smart Call Home messages that are 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 the switch generates a Smart Call Home message to all e-mail addresses in the destination profile. The switch 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 Nexus switches support the following predefined destination profiles:

- CiscoTAC-1—Supports the Cisco-TAC alert group in XML message format.
- 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 switch sends Smart Call Home alerts to e-mail 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 7: 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. |
| Diagnostic | Events generated by diagnostics. | **show diagnostic result module all detail**<br>**show moduleshow version**<br>**show tech-support platform callhome** |
| Supervisor hardware | Events related to supervisor modules. | **show diagnostic result module all detail**<br>**show moduleshow version**<br>**show tech-support platform callhome** |
| Linecard hardware | Events related to standard or intelligent switching modules. | **show diagnostic result module all detail**<br>**show moduleshow version**<br>**show tech-support platform callhome** |
| Configuration | Periodic events related to configuration. | **show version**<br>**show module**<br>**show running-config all**<br>**show startup-config** |
| System | Events generated by a failure of a software system that is critical to unit operation. | **show system redundancy status**<br>**show tech-support** |
| Environmental | Events related to power, fan, and environment-sensing elements such as temperature alarms. | **show environment**<br>**show logging last 1000**<br>**show module show version**<br>**show tech-support platform callhome** |

| 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 module**<br>**show version**<br>**show license usage**<br>**show inventory**<br>**show sprom all**<br>**show system uptime** |

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 **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 destination profile (predefined and user defined) with a Smart Call Home message level threshold. The switch does not generate any Smart Call Home messages with a value lower than this threshold for the destination profile. The Smart Call Home message level ranges from 0 (lowest level of urgency) to 9 (highest level of urgency), and the default is 0 (the switch sends all messages).

Smart Call Home messages that are sent for syslog alert groups have the syslog severity level mapped to the Smart Call Home message level.

**Note** Smart Call Home does not change the syslog message level in the message text.

The following table shows each Smart Call Home message level keyword and the corresponding syslog level for the syslog port alert group.

***Table 8: 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. |

| Smart Call Home Level | Keyword | Syslog Level | Description |
|---|---|---|---|
| 4 | Minor | Error (3) | Minor conditions. |
| 3 | Warning | Warning (4) | Warning conditions. |
| 2 | Notification | Notice (5) | Basic notification and informational messages. |
| 1 | Normal | Information (6) | Normal event signifying return to normal state. |
| 0 | Debugging | Debug (7) | Debugging messages. |

# Call Home Message Formats

Call Home supports the following message formats:

- Short text message format
- Common fields for all full text and XML messages
- Inserted fields for a reactive or proactive event message
- Inserted fields for an inventory event message
- Inserted fields for a user-generated test message

The following table describes the short text formatting option for all message types.

***Table 9: Short Text Message Format***

| 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 |
| Alarm urgency level | Error level such as that applied to a system message |

The following table describes the common event message format for full text or XML.

***Table 10: Common Fields for All Full Text and XML Messages***

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|---|---|---|
| Time stamp | Date and time stamp of event in ISO time notation:<br>*YYYY-MM-DD HH:MM:SS GMT+HH:MM* | /aml/header/time |

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|---|---|---|
| Message name | Name of message. Specific event names are listed in the preceding table. | /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. | /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 *type@Sid@serial*:<br>• *type* is the product model number from backplane IDPROM.<br>• @ is a separator character.<br>• *Sid* is C, identifying the serial ID as a chassis serial number.<br>• *serial* is the number identified by the Sid field.<br>An example is WS-C6509@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 |
| Site ID | Optional user-configurable field used for Cisco-supplied site ID or other data meaningful to alternate support service. | /aml/ header/siteID |

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|---|---|---|
| Server ID | If the message is generated from the device, this is the unique device identifier (UDI) of the device.<br>The format is *type@Sid@serial*:<br>• *type* is the product model number from backplane IDPROM.<br>• @ is a separator character.<br>• *Sid* is C, identifying the serial ID as a chassis serial number.<br>• *serial* is the number identified by the Sid field.<br>An example is WS-C6509@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 e-mail | E-mail 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/sysContactPhoneNumber |
| 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 |
| Fields specific to a particular alert group message are inserted here. | | |
| The following fields may be repeated if multiple CLI commands are executed for this 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 |

The following table describes the reactive event message format for full text or XML.

*Table 11: Inserted Fields for a Reactive or Proactive Event Message*

| 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 |

The following table describes the inventory event message format for full text or XML.

*Table 12: Inserted Fields for an Inventory Event Message*

| 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 |
| FRU s/n | Serial number of the FRU. | /aml/body/fru/serialNo |
| FRU part number | Part number of the FRU. | /aml/body/fru/partNo |

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|---|---|---|
| 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 |

The following table describes the user-generated test message format for full text or XML.

***Table 13: Inserted Fields for a User-Generated Test Message***

| 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 |

# Guidelines and Limitations for Smart Call Home

- 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 switch cannot send Smart Call Home messages.
- Operates with any SMTP e-mail server.

**Note** Starting with Release 7.0(3)I2(1), the SNMP syscontact is not configured by default. You have to explicitly use the **snmp-server contact** <*sys-contact*> command to configure the SNMP syscontact. When this command is configured, the feature callhome gets enabled.

# Prerequisites for Smart Call Home

- You must have e-mail server connectivity.
- You must have access to contact name (SNMP server contact), phone, and street address information.
- You must have IP connectivity between the switch and the e-mail server.
- You must have an active service contract for the device that you are configuring.

# Default Call Home Settings

*Table 14: Default Call Home Parameters*

| Parameters | Default |
|---|---|
| Destination message size for a message sent in full text format | 4000000 |
| Destination message size for a message sent in XML format | 4000000 |
| Destination message size for a message sent in short text format | 4000 |
| SMTP server port number if no port is specified | 25 |
| 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 |
| Call Home message level | 0 (zero) |

# Configuring Smart Call Home

## Registering for Smart Call Home

### Before you begin

- Know the sMARTnet contract number for your switch
- Know your e-mail address
- Know your Cisco.com ID

### Procedure

**Step 1** In a browser, navigate to the Smart Call Home web page:

http://www.cisco.com/go/smartcall/

**Step 2** Under **Getting Started**, follow the directions to register Smart Call Home.

### What to do next

Configure contact information.

# Configuring Contact Information

You must configure the e-mail, phone, and street address information for Smart Call Home. You can optionally configure the contract ID, customer ID, site ID, and switch priority information.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **snmp-server contact** *sys-contact* | Configures the SNMP sysContact. |
| **Step 3** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |
| **Step 4** | switch(config-callhome)# **email-contact** *email-address* | Configures the e-mail address for the primary person responsible for the switch.<br>The *email-address* can be up to 255 alphanumeric characters in an e-mail address format.<br>**Note**<br>You can use any valid e-mail address. The address cannot contain spaces. |
| **Step 5** | switch(config-callhome)# **phone-contact** *international-phone-number* | Configures the phone number in international phone number format for the primary person responsible for the device. The *international-phone-number* can be up to 17 alphanumeric characters and must be in international phone number format.<br>**Note**<br>The phone number cannot contain spaces. Use the plus (+) prefix before the number. |
| **Step 6** | switch(config-callhome)# **streetaddress** *address* | Configures the street address for the primary person responsible for the switch.<br>The *address* can be up to 255 alphanumeric characters. Spaces are accepted. |
| **Step 7** | (Optional) switch(config-callhome)# **contract-id** *contract-number* | Configures the contract number for this switch from the service agreement.<br>The *contract-number* can be up to 255 alphanumeric characters. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 8** | (Optional) switch(config-callhome)# **customer-id** *customer-number* | Configures the customer number for this switch from the service agreement.<br>The *customer-number* can be up to 255 alphanumeric characters. |
| **Step 9** | (Optional) switch(config-callhome)# **site-id** *site-number* | Configures the site number for this switch.<br>The *site-number* can be up to 255 alphanumeric characters in free format. |
| **Step 10** | (Optional) switch(config-callhome)# **switch-priority** *number* | Configures the switch priority for this switch.<br>The range is from 0 to 7, with 0 being the highest priority and 7 the lowest. The default is 7. |
| **Step 11** | (Optional) switch# **show callhome** | Displays a summary of the Smart Call Home configuration. |
| **Step 12** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure the contact information for Call Home:

```
switch# configuration terminal
switch(config)# snmp-server contact personname@companyname.com
switch(config)# callhome
switch(config-callhome)# email-contact personname@companyname.com
switch(config-callhome)# phone-contact +1-800-123-4567
switch(config-callhome)# street-address 123 Anystreet St., Anycity, Anywhere
```

### What to do next

Create a destination profile.

# Creating a Destination Profile

You must create a user-defined destination profile and configure the message format for that new destination profile.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-callhome)# **destination-profile** {**ciscoTAC-1** {**alert-group** *group* \| **email-addr** *address* \| **http** *URL* \| **transport-method** {**email** \| **http**}} \| *profilename* {**alert-group** *group* \| **email-addr** *address* \| **format** {**XML** \| **full-txt** \| **short-txt**} \| **http** *URL* \| **message-level** *level* \| **message-size** *size* \| **transport-method** {**email** \| **http**}} \| **full-txt-destination** {**alert-group** *group* \| **email-addr** *address* \| **http** *URL* \| **message-level** *level* \| **message-size** *size* \| **transport-method** {**email** \| **http**}} \| **short-txt-destination** {**alert-group** *group* \| **email-addr** *address* \| **http** *URL* \| **message-level** *level* \| **message-size** *size* \| **transport-method** {**email** \| **http**}}} | Creates a new destination profile and sets the message format for the profile. The profile-name can be any alphanumeric string up to 31 characters.<br>For further details about this command, see the command reference for your platform. |
| **Step 4** | (Optional) switch# **show callhome destination-profile** [**profile** *name*] | Displays information about one or more destination profiles. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to create a destination profile for Smart Call Home:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile Noc101 format full-text
```

# Modifying a Destination Profile

You can modify the following attributes for a predefined or user-defined destination profile:

- Destination address—The actual address, pertinent to the transport mechanism, to which the alert should be sent.
- Message formatting—The message format used for sending the alert (full text, short text, or XML).
- Message level—The Call Home message severity level for this destination profile.
- Message size—The allowed length of a Call Home message sent to the e-mail addresses in this destination profile.

**Note** You cannot modify or delete the CiscoTAC-1 destination profile.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |
| **Step 3** | switch(config-callhome)# **destination-profile** {*name* \| **full-txt-destination** \| **short-txt-destination**} **email-addr** *address* | Configures an e-mail address for a user-defined or predefined destination profile. You can configure up to 50 e-mail addresses in a destination profile. |
| **Step 4** | **destination-profile** {*name* \| **full-txt-destination** \| **short-txt-destination**} **message-level** *number* | Configures the Smart Call Home message severity level for this destination profile. The switch sends only alerts that have a matching or higher Smart Call Home severity level to destinations in this profile. The range for the *number* is from 0 to 9, where 9 is the highest severity level. |
| **Step 5** | switch(config-callhome)# **destination-profile** {*name* \| **full-txt-destination** \| **short-txt-destination**} **message-size** *number* | Configures the maximum message size for this destination profile. The range is from 0 to 5000000 for full-txt-destination and the default is 2500000. The range is from 0 to 100000 for short-txt-destination and the default is 4000. The value is 5000000 for CiscoTAC-1, which is not changeable. |
| **Step 6** | (Optional) switch# **show callhome destination-profile** [**profile** *name*] | Displays information about one or more destination profiles. |
| **Step 7** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to modify a destination profile for Smart Call Home:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile full-text-destination email-addr
person@example.com
switch(config-callhome)# destination-profile full-text-destination message-level 5
switch(config-callhome)# destination-profile full-text-destination message-size 10000
switch(config-callhome)#
```

### What to do next

Associate an alert group with a destination profile.

# Associating an Alert Group with a Destination Profile

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |
| **Step 3** | switch(config-callhome)# **destination-profile** *name* **alert-group** {**All** \| **Cisco-TAC** \| **Configuration** \| **Diagnostic** \| **Environmental** \| **Inventory** \| **License** \| **Linecard-Hardware** \| **Supervisor-Hardware** \| **Syslog-group-port** \| **System** \| **Test**} | Associates an alert group with this destination profile. Use the **All** keyword to associate all alert groups with the destination profile. |
| **Step 4** | (Optional) switch# **show callhome destination-profile** [**profile** *name*] | Displays information about one or more destination profiles. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to associate all alert groups with the destination profile Noc101:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile Noc101 alert-group All
switch(config-callhome)#
```

### What to do next

Optionally, you can add **show** commands to an alert group and configure the SMTP e-mail server.

# Adding Show Commands to an Alert Group

You can assign a maximum of five user-defined **show** commands to an alert group.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-callhome)# **alert-group** {**Configuration** \| **Diagnostic** \| **Environmental** \| **Inventory** \| **License** \| **Linecard-Hardware** \| **Supervisor-Hardware** \| **Syslog-group-port** \| **System** \| **Test**} **user-def-cmd** *show-cmd* | Adds the **show** command output to any Call Home messages sent for this alert group. Only valid **show** commands are accepted.<br>**Note** You cannot add user-defined **show** commands to the CiscoTAC-1 destination profile. |
| **Step 4** | (Optional) switch# **show callhome user-def-cmds** | Displays information about all user-defined **show** commands added to alert groups. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to add the **show ip routing** command to the Cisco-TAC alert group:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# alert-group Configuration user-def-cmd show ip routing
switch(config-callhome)#
```

### What to do next

Configure Smart Call Home to connect to the SMTP e-mail server.

# Configuring E-Mail Server Details

You must configure the SMTP server address for the Smart Call Home functionality to work. You can also configure the from and reply-to e-mail addresses.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |
| **Step 3** | switch(config-callhome)# **transport email smtp-server** *ip-address* [**port** *number*] [**use-vrf** *vrf-name*] | Configures the SMTP server as either the domain name server (DNS) name, IPv4 address, or IPv6 address.<br>The *number* range is from1 to 65535. The default port number is 25. |

| | Command or Action | Purpose |
|---|---|---|
| | | Optionally, you can configure the VRF instance to use when communicating with this SMTP server. |
| **Step 4** | (Optional) switch(config-callhome)# **transport email from** *email-address* | Configures the e-mail from field for Smart Call Home messages. |
| **Step 5** | (Optional) switch(config-callhome)# **transport email reply-to** *email-address* | Configures the e-mail reply-to field for Smart Call Home messages. |
| **Step 6** | (Optional) switch# **show callhome transport-email** | Displays information about the e-mail configuration for Smart Call Home. |
| **Step 7** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure the e-mail options for Smart Call Home messages:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# transport email smtp-server 192.0.2.10 use-vrf Red
switch(config-callhome)# transport email from person@example.com
switch(config-callhome)# transport email reply-to person@example.com
switch(config-callhome)#
```

### What to do next

Configure periodic inventory notifications.

# Configuring Periodic Inventory Notifications

You can configure the switch to periodically send a message with an inventory of all software services currently enabled and running on the device with hardware inventory information. The switch generates two Smart Call Home notifications; periodic configuration messages and periodic inventory messages.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |
| **Step 3** | switch(config-callhome)# **periodic-inventory notification** [**interval** *days*] [**timeofday** *time*] | Configures periodic inventory messages.<br>The **interval** *days* range is from 1 to 30 days.<br>The default is 7 days. |

| | Command or Action | Purpose |
|---|---|---|
| | | The **timeofday** *time* is in HH:MM format. |
| **Step 4** | (Optional) switch# **show callhome** | Displays information about Smart Call Home. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure the periodic inventory messages to generate every 20 days:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# periodic-inventory notification interval 20
switch(config-callhome)#
```

### What to do next

Disable duplicate message throttling.

# Disabling Duplicate Message Throttling

You can limit the number of duplicate messages received for the same event. By default, the switch 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 switch discards further messages for that alert type.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |
| **Step 3** | switch(config-callhome) # **no duplicate-message throttle** | Disables duplicate message throttling for Smart Call Home.<br>Duplicate message throttling is enabled by default. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to disable duplicate message throttling:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# no duplicate-message throttle
switch(config-callhome)#
```

### What to do next

Enable Smart Call Home.

# Enabling or Disabling Smart Call Home

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |
| **Step 3** | switch(config-callhome) # [**no**] **enable** | Enables or disables Smart Call Home.<br>Smart Call Home is disabled by default. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to enable Smart Call Home:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# enable
switch(config-callhome)#
```

### What to do next

Optionally, generate a test message.

# Testing the Smart Call Home Configuration

### Before you begin

Verify that the message level for the destination profile is set to 2 or lower.

**Important** Smart Call Home testing fails when the message level for the destination profile is set to 3 or higher.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **callhome** | Enters Smart Call Home configuration mode. |
| **Step 3** | switch(config-callhome) # **callhome send diagnostic** | Sends the specified Smart Call Home message to all configured destinations. |
| **Step 4** | switch(config-callhome) # **callhome test** | Sends a test message to all configured destinations. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

#### Example

The following example shows how to enable Smart Call Home:

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# callhome send diagnostic
switch(config-callhome)# callhome test
switch(config-callhome)#
```

# Verifying the Smart Call Home Configuration

Use one of the following commands to verify the configuration:

| Command | Purpose |
|---|---|
| **show callhome** | Displays the status for Smart Call Home. |
| **show callhome destination-profile** *name* | Displays one or more Smart Call Home destination profiles. |
| **show callhome pending-diff** | Displays the differences between he pending and running Smart Call Home configuration. |
| **show callhome status** | Displays the Smart Call Home status. |
| **show callhome transport-email** | Displays the e-mail configuration for Smart Call Home. |
| **show callhome user-def-cmds** | Displays CLI commands added to any alert groups. |
| **show running-config** [**callhome** \| **callhome-all**] | Displays the running configuration for Smart Call Home. |
| **show startup-config callhome** | Displays the startup configuration for Smart Call Home. |

| Command | Purpose |
|---|---|
| **show tech-support callhome** | Displays the technical support output for Smart Call Home. |

# Sample Syslog Alert Notification in Full-Text Format

This sample shows the full-text format for a syslog port alert-group notification:

```
source:MDS9000
Switch Priority:7
Device Id:WS-C6509@C@FG@07120011
Customer Id:Example.com
Contract Id:123
Site Id:San Jose
Server Id:WS-C6509@C@FG@07120011
Time of Event:2004-10-08T11:10:44
Message Name:SYSLOG_ALERT
Message Type:Syslog
Severity Level:2
System Name:10.76.100.177
Contact Name:User Name
Contact Email:person@example.com
Contact Phone:+1-408-555-1212
Street Address:#1234 Any Street, Any City, Any State, 12345
Event Description:2006 Oct  8 11:10:44 10.76.100.177 %PORT-5-IF_TRUNK_UP:
%$VLAN 1%$ Interface e2/5, vlan 1 is up
syslog_facility:PORT
start chassis information:
Affected Chassis:WS-C6509
Affected Chassis Serial Number:FG@07120011
Affected Chassis Hardware Version:0.104
Affected Chassis Software Version:3.1(1)
Affected Chassis Part No:73-8607-01
end chassis information:
```

# Sample Syslog Alert Notification in XML Format

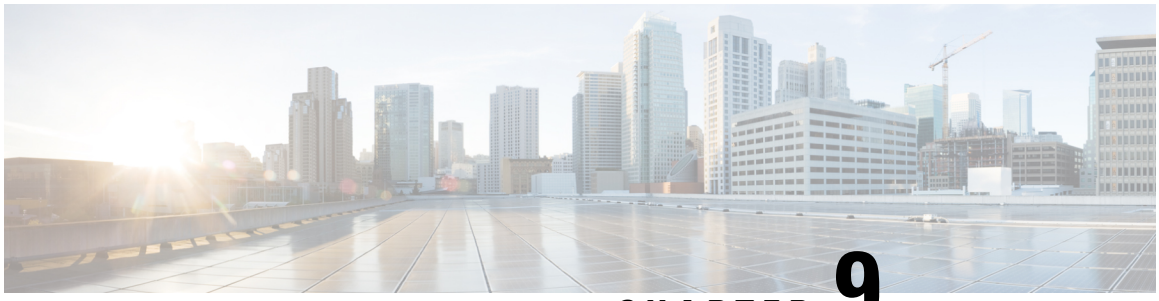
This sample shows the XML format for a syslog port alert-group notification:

```
From: example
Sent: Wednesday, April 25, 2007 7:20 AM
To: User (user)
Subject: System Notification From Router - syslog - 2007-04-25 14:19:55
GMT+00:00
<?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.example.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.example.com/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.example.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.example.com/appliance/uri</aml-session:From>
<aml-session:MessageId>M2:69000101:C9D9E20B</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
```

```
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.example.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.example.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2007-04-25 14:19:55 GMT+00:00</aml-block:CreationDate>
<aml-block:Builder>
<aml-block:Name>Cat6500</aml-block:Name>
<aml-block:Version>2.0</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>G3:69000101:C9F9E20C</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>2</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:Call Home xmlns:ch="http://www.example.com/2005/05/callhome" version="1.0">
<ch:EventTime>2007-04-25 14:19:55 GMT+00:00</ch:EventTime>
<ch:MessageDescription>03:29:29: %CLEAR-5-COUNTERS: Clear counter on all
interfaces by console</ch:MessageDescription>
<ch:Event>
<ch:Type>syslog</ch:Type>
<ch:SubType>
</ch:SubType>
<ch:Brand>Cisco Systems</ch:Brand>
<ch:Series>Catalyst 6500 Series Switches</ch:Series>
</ch:Event>
<ch:CustomerData>
<ch:UserData>
<ch:Email>person@example.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:CustomerId>12345</ch:CustomerId>
<ch:SiteId>building 1</ch:SiteId>
<ch:ContractId>abcdefg12345</ch:ContractId>
<ch:DeviceId>WS-C6509@C@69000101</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>Router</ch:Name>
<ch:Contact>
</ch:Contact>
<ch:ContactEmail>user@example.com</ch:ContactEmail>
<ch:ContactPhoneNumber>+1-408-555-1212</ch:ContactPhoneNumber>
<ch:StreetAddress>#1234 Any Street, Any City, Any State, 12345
</ch:StreetAddress>
</ch:SystemInfo>
</ch:CustomerData>
<ch:Device>
<rme:Chassis xmlns:rme="http://www.example.com/rme/4.0">
<rme:Model>WS-C6509</rme:Model>
<rme:HardwareVersion>1.0</rme:HardwareVersion>
<rme:SerialNumber>69000101</rme:SerialNumber>
<rme:AdditionalInformation>
<rme:AD name="PartNumber" value="73-3438-03 01" />
<rme:AD name="SoftwareVersion" value="4.0(20080421:012711)" />
</rme:AdditionalInformation>
</rme:Chassis>
</ch:Device>
</ch:Call Home>
</aml-block:Content>
<aml-block:Attachments>
```

```
<aml-block:Attachment type="inline">
<aml-block:Name>show logging</aml-block:Name>
<aml-block:Data encoding="plain">
<![CDATA[Syslog logging: enabled (0 messages dropped, 0 messages
rate-limited, 0 flushes, 0 overruns, xml disabled, filtering disabled)
    Console logging: level debugging, 53 messages logged, xml disabled,
filtering disabled    Monitor logging: level debugging, 0 messages logged,
xml disabled,filtering disabled    Buffer logging: level debugging,
53 messages logged, xml disabled,      filtering disabled    Exception
Logging: size (4096 bytes)    Count and timestamp logging messages: disabled
    Trap logging: level informational, 72 message lines logged
Log Buffer (8192 bytes):
00:00:54: curr is 0x20000
00:00:54: RP: Currently running ROMMON from F2 region
00:01:05: %SYS-5-CONFIG_I: Configured from memory by console
00:01:09: %SYS-5-RESTART: System restarted --Cisco IOS Software,
s72033_rp Software (s72033_rp-ADVENTERPRISEK9_DBG-VM), Experimental
Version 12.2(20070421:012711) Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-07 15:54 by xxx
Firmware compiled 11-Apr-07 03:34 by integ Build [100]00:01:01: %PFREDUN-6-ACTIVE:
 Initializing as ACTIVE processor for this switch00:01:01: %SYS-3-LOGGER_FLUSHED:
System was paused for 00:00:00 to ensure console debugging output.00:03:00: SP: SP:
 Currently running ROMMON from F1 region00:03:07: %C6K_PLATFORM-SP-4-CONFREG_BREAK
_ENABLED: The default factory setting for config register is 0x2102.It is advisable
 to retain 1 in 0x2102 as it prevents returning to ROMMON when break is issued.00:03:18:
 %SYS-SP-5-RESTART: System restarted --Cisco IOS Software, s72033_sp Software
 (s72033_sp-ADVENTERPRISEK9_DBG-VM), Experimental Version 12.2(20070421:012711)Copyright
 (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-07 18:00 by xxx
00:03:18: %SYS-SP-6-BOOTTIME: Time taken to reboot after reload =  339 seconds
00:03:18: %OIR-SP-6-INSPS: Power supply inserted in slot 1
00:03:18: %C6KPWR-SP-4-PSOK: power supply 1 turned on.
00:03:18: %OIR-SP-6-INSPS: Power supply inserted in slot00:01:09: %SSH-5-ENABLED:
 SSH 1.99 has been enabled
00:03:18: %C6KPWR-SP-4-PSOK: power supply 2 turned on.
00:03:18: %C6KPWR-SP-4-PSREDUNDANTMISMATCH: power supplies rated outputs do not match.
00:03:18: %C6KPWR-SP-4-PSREDUNDANTBOTHSUPPLY: in power-redundancy mode, system is
 operating on both power supplies.
00:01:10: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
00:01:10: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
00:03:20: %C6KENV-SP-4-FANHIOUTPUT: Version 2 high-output fan-tray is in effect
00:03:22: %C6KPWR-SP-4-PSNOREDUNDANCY: Power supplies are not in full redundancy,
 power usage exceeds lower capacity supply
00:03:26: %FABRIC-SP-5-FABRIC_MODULE_ACTIVE: The Switch Fabric Module in slot 6
 became active.
00:03:28: %DIAG-SP-6-RUN_MINIMUM: Module 6: Running Minimal Diagnostics...
00:03:50: %DIAG-SP-6-DIAG_OK: Module 6: Passed Online Diagnostics
00:03:50: %OIR-SP-6-INSCARD: Card inserted in slot 6, interfaces are now online
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 3: Running Minimal Diagnostics...
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 7: Running Minimal Diagnostics...
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 9: Running Minimal Diagnostics...
00:01:51: %MFIB_CONST_RP-6-REPLICATION_MODE_CHANGE: Replication Mode Change Detected.
 Current system replication mode is Ingress
00:04:01: %DIAG-SP-6-DIAG_OK: Module 3: Passed Online Diagnostics
00:04:01: %OIR-SP-6-DOWNGRADE: Fabric capable module 3 not at an appropriate hardware
 revision level, and can only run in flowthrough mode
00:04:02: %OIR-SP-6-INSCARD: Card inserted in slot 3, interfaces are now online
00:04:11: %DIAG-SP-6-DIAG_OK: Module 7: Passed Online Diagnostics
00:04:14: %OIR-SP-6-INSCARD: Card inserted in slot 7, interfaces are now online
00:04:35: %DIAG-SP-6-DIAG_OK: Module 9: Passed Online Diagnostics
00:04:37: %OIR-SP-6-INSCARD: Card inserted in slot 9, interfaces are now online
00:00:09: DaughterBoard (Distributed Forwarding Card 3)
Firmware compiled 11-Apr-07 03:34 by integ Build [100]
00:00:22: %SYS-DFC4-5-RESTART: System restarted --
```

```
Cisco DCOS Software, c6lc2 Software (c6lc2-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 17:20 by xxx
00:00:23: DFC4: Currently running ROMMON from F2 region
00:00:25: %SYS-DFC2-5-RESTART: System restarted --
Cisco IOS Software, c6slc Software (c6slc-SPDBG-VM), Experimental Version 12.2
(20070421:012711)Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 16:40 by username1
00:00:26: DFC2: Currently running ROMMON from F2 region
00:04:56: %DIAG-SP-6-RUN_MINIMUM: Module 4: Running Minimal Diagnostics...
00:00:09: DaughterBoard (Distributed Forwarding Card 3)
Firmware compiled 11-Apr-08 03:34 by integ Build [100]
slot_id is 8
00:00:31: %FLASHFS_HES-DFC8-3-BADCARD: /bootflash:: The flash card seems to
 be corrupted
00:00:31: %SYS-DFC8-5-RESTART: System restarted --
Cisco DCOS Software, c6lc2 Software (c6lc2-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 17:20 by username1
00:00:31: DFC8: Currently running ROMMON from S (Gold) region
00:04:59: %DIAG-SP-6-RUN_MINIMUM: Module 2: Running Minimal Diagnostics...
00:05:12: %DIAG-SP-6-RUN_MINIMUM: Module 8: Running Minimal Diagnostics...
00:05:13: %DIAG-SP-6-RUN_MINIMUM: Module 1: Running Minimal Diagnostics...
00:00:24: %SYS-DFC1-5-RESTART: System restarted --
Cisco DCOS Software, c6slc Software (c6slc-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 16:40 by username1
00:00:25: DFC1: Currently running ROMMON from F2 region
00:05:30: %DIAG-SP-6-DIAG_OK: Module 4: Passed Online Diagnostics
00:05:31: %SPAN-SP-6-SPAN_EGRESS_REPLICATION_MODE_CHANGE: Span Egress HW
Replication Mode Change Detected. Current replication mode for unused asic
 session 0 is Centralized
00:05:31: %SPAN-SP-6-SPAN_EGRESS_REPLICATION_MODE_CHANGE: Span Egress HW
Replication Mode Change Detected. Current replication mode for unused asic
 session 1 is Centralized
00:05:31: %OIR-SP-6-INSCARD: Card inserted in slot 4, interfaces are now online
00:06:02: %DIAG-SP-6-DIAG_OK: Module 1: Passed Online Diagnostics
00:06:03: %OIR-SP-6-INSCARD: Card inserted in slot 1, interfaces are now online
00:06:31: %DIAG-SP-6-DIAG_OK: Module 2: Passed Online Diagnostics
00:06:33: %OIR-SP-6-INSCARD: Card inserted in slot 2, interfaces are now online
00:04:30: %XDR-6-XDRIPCNOTIFY: Message not sent to slot 4/0 (4) because of IPC
 error timeout. Disabling linecard. (Expected during linecard OIR)
00:06:59: %DIAG-SP-6-DIAG_OK: Module 8: Passed Online Diagnostics
00:06:59: %OIR-SP-6-DOWNGRADE_EARL: Module 8 DFC installed is not identical to
 system PFC and will perform at current system operating mode.
00:07:06: %OIR-SP-6-INSCARD: Card inserted in slot 8, interfaces are now online
Router#]]>
</aml-block:Data>
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>
```



CHAPTER 9

# Configuring Session Manager

This chapter contains the following sections:


- Information About Session Manager, on page 113
- Guidelines and Limitations for Session Manager, on page 113
- Configuring Session Manager, on page 114
- Verifying the Session Manager Configuration, on page 116


## Information 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 implements the changes atomically 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.

## Guidelines and Limitations for Session Manager

Session Manager has the following configuration guidelines and limitations:

- Session Manager supports only the access control list (ACL) feature.
- You can create up to 32 configuration sessions.
- You can configure a maximum of 20,000 commands across all sessions.

# Configuring Session Manager

## Creating a Session

You can create up to 32 configuration sessions.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure session** *name* | Creates a configuration session and enters session configuration mode. The name can be any alphanumeric string.<br>Displays the contents of the session. |
| **Step 2** | (Optional) switch(config-s)# **show configuration session** [*name*] | Displays the contents of the session. |
| **Step 3** | (Optional) switch(config-s)# **save** *location* | Saves the session to a file. The location can be in bootflash or volatile. |

## Configuring ACLs in a Session

You can configure ACLs within a configuration session.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure session** *name* | Creates a configuration session and enters session configuration mode. The name can be any alphanumeric string. |
| **Step 2** | switch(config-s)# **ip access-list** *name* | Creates an ACL. |
| **Step 3** | (Optional) switch(config-s-acl)# **permit** *protocol source destination* | Adds a permit statement to the ACL. |
| **Step 4** | switch(config-s-acl)# **interface** *interface-type number* | Enters interface configuration mode. |
| **Step 5** | switch(config-s-if)# **ip port access-group** *name* **in** | Adds a port access group to the interface. |
| **Step 6** | (Optional) switch# **show configuration session** [*name*] | Displays the contents of the session. |

# Verifying a Session

To verify a session, use the following command in session mode:

| Command | Purpose |
|---|---|
| switch(config-s)# **verify** [**verbose**] | Verifies the commands in the configuration session. |

# Committing a Session

To commit a session, use the following command in session mode:

| Command | Purpose |
|---|---|
| switch(config-s)# **commit** [**verbose**] | Commits the commands in the configuration session. |

# Saving a Session

To save a session, use the following command in session mode:

| Command | Purpose |
|---|---|
| switch(config-s)# **save** *location* | (Optional) Saves the session to a file. The location can be in bootflash or volatile. |

# Discarding a Session

To discard a session, use the following command in session mode:

| Command | Purpose |
|---|---|
| switch(config-s)# **abort** | Discards the configuration session without applying the commands. |

# Configuration Example for Session Manager

The following example shows how to create a configuration session for ACLs:

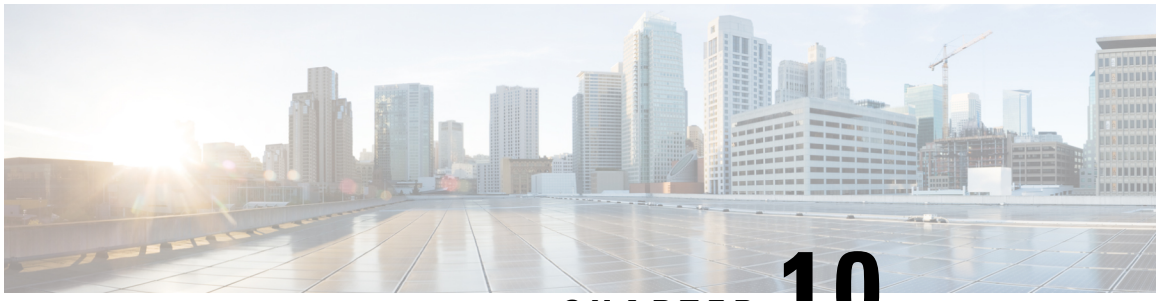
```
switch# configure session name test2
switch(config-s)# ip access-list acl2
switch(config-s-acl)# permit tcp any any
switch(config-s-acl)# exit
switch(config-s)# interface Ethernet 1/4
switch(config-s-ip)# ip port access-group acl2 in
switch(config-s-ip)# exit
switch(config-s)# verify
switch(config-s)# exit
```

```
switch# show configuration session test2
```

# Verifying the Session Manager Configuration

To verify 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. |



CHAPTER 10

# Configuring the Scheduler

This chapter contains the following sections:


- Information About the Scheduler, on page 117
- Guidelines and Limitations for the Scheduler, on page 118
- Default Settings for the Scheduler, on page 118
- Configuring the Scheduler, on page 119
- Verifying the Scheduler Configuration, on page 125
- Configuration Examples for the Scheduler, on page 125
- Standards for the Scheduler, on page 126


## Information About the Scheduler

The scheduler allows you to define and set a timetable for maintenance activities such as the following:

- Quality of service 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— Job is completed once a day.
  - Weekly— Job is completed once a week.

- Monthly—Job is completed once a month.
- Delta—Job begins at the specified start time and then at specified intervals (days:hours:minutes).

- One-time mode—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. Because user credentials from a remote authentication are not retained long enough to support a scheduled job, you must 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.

## Scheduler Log Files

The scheduler maintains a log file that contains the job output. If the size of the job output is greater than the size of the log file, the output is truncated.

# Guidelines and Limitations for the Scheduler

- The scheduler can fail if it encounters one of the following while performing a job:
  - If a feature license is expired when a job for that feature is scheduled.
  - If a feature is disabled at the time when a job for that feature is scheduled.
- Verify that you have configured the time. The scheduler does not apply a default timetable. If you create a schedule, 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.

# Default Settings for the Scheduler

***Table 15: Default Command Scheduler Parameters***

| Parameters | Default |
|---|---|
| Scheduler state | Disabled |
| Log file size | 16 KB |

# Configuring the Scheduler

## Enabling the Scheduler

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **feature scheduler** | Enables the scheduler. |
| **Step 3** | (Optional) switch(config) # **show scheduler config** | Displays the scheduler configuration. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to enable the scheduler:

```
switch# configure terminal
switch(config)# feature scheduler
switch(config)# show scheduler config
config terminal
   feature scheduler
   scheduler logfile size 16
end
switch(config)#
```

## Defining the Scheduler Log File Size

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **scheduler logfile size** *value* | Defines the scheduler log file size in kilobytes.<br>The range is from 16 to 1024. The default log file size is 16.<br>**Note**<br>If the size of the job output is greater than the size of the log file, the output is truncated. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to define the scheduler log file size:

```
switch# configure terminal
switch(config)# scheduler logfile size 1024
switch(config)#
```

# Configuring Remote User Authentication

Remote users must authenticate with their clear text password before creating and configuring jobs.

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** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **scheduler aaa-authentication password** [**0** \| **7**] *password* | Configures a password for the user who is currently logged in.<br>To configure a clear text password, enter **0**.<br>To configure an encrypted password, enter **7**. |
| **Step 3** | switch(config) # **scheduler aaa-authentication username** *name* **password** [**0** \| **7**] *password* | Configures a clear text password for a remote user. |
| **Step 4** | (Optional) switch(config) # **show running-config \| include "scheduler aaa-authentication"** | Displays the scheduler password information. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure a clear text password for a remote user called NewUser:

```
switch# configure terminal
switch(config) # scheduler aaa-authentication
username NewUser password z98y76x54b
```

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

# Defining a Job

After you define a job, you cannot modify or remove commands. To change the job, delete it and create a new one.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **scheduler job name** *name* | Creates a job with the specified name and enters the job configuration mode.<br>The *name* is restricted to 31 characters. |
| **Step 3** | switch(config-job) # *command1* **;** [*command2* **;***command3* **;** ... | Defines the sequence of commands for the specified job. Separate commands with spaces and semicolons (;).<br>Create the filename using the current timestamp and switch name. |
| **Step 4** | (Optional) switch(config-job) # **show scheduler job** [*name*] | Displays the job information.<br>The *name* is restricted to 31 characters. |
| **Step 5** | (Optional) switch(config-job) # **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to:

- Create a scheduler job named "backup-cfg"
- Save the running configuration to a file in the bootflash
- Copy the file from the bootflash to a TFTP server
- Save the change to the startup configuration

```
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) # copy running-config startup-config
```

# Deleting a Job

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **no scheduler job name** *name* | Deletes the specified job and all commands defined within it.<br>The *name* is restricted to 31 characters. |
| **Step 3** | (Optional) switch(config-job) # **show scheduler job** [*name*] | Displays the job information. |
| **Step 4** | (Optional) switch(config-job) # **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to delete a job called configsave:

```
switch# configure terminal
switch(config)# no scheduler job name configsave
switch(config-job)# copy running-config startup-config
switch(config-job)#
```

# Defining a Timetable

You must configure a timetable. Otherwise, jobs will not be scheduled.

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, 2008, 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, 2008, 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** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **scheduler schedule name** *name* | Creates a new scheduler and enters schedule configuration mode for that schedule.<br>The *name* is restricted to 31 characters. |
| **Step 3** | switch(config-schedule) # **job name** *name* | Associates a job with this schedule. You can add multiple jobs to a schedule.<br>The *name* is restricted to 31 characters. |
| **Step 4** | switch(config-schedule) # **time daily** *time* | Indicates the job starts every day at a designated time, specified as HH:MM. |
| **Step 5** | switch(config-schedule) # **time weekly** [[*day-of-week*:] *HH*:] *MM* | Indicates that the job starts on a specified day of the week.<br>The day of the week is represented by an integer (for example, **1** for Sunday, **2** for Monday) or as an abbreviation (for example, **sun**, **mon**).<br>The maximum length for the entire argument is 10 characters. |
| **Step 6** | switch(config-schedule) # **time monthly** [[*day-of-month*:] *HH*:] *MM* | Indicates that the job starts on a specified day each month.<br>If you specify 29, 30, or 31, the job is started on the last day of each month. |
| **Step 7** | switch(config-schedule) # **time start** {**now repeat** *repeat-interval* \| *delta-time* [**repeat** *repeat-interval*]} | Indicates the job starts periodically.<br>The start-time format is [[[[yyyy:]mmm:]dd:]HH]:MM.<br>• *delta-time*— Specifies the amount of time to wait after the schedule is configured before starting a job.<br>• **now**— Specifies that the job starts two minutes from now.<br>• **repeat** *repeat-interval*— Specifies the frequency at which the job is repeated. |
| **Step 8** | (Optional) switch(config-schedule) # **show scheduler config** | Displays the scheduler information. |
| **Step 9** | (Optional) switch(config-schedule) # **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to define a timetable where jobs start on the 28th of each month at 23:00 hours:

```
switch# configure terminal
switch(config)# scheduler schedule name weekendbackupqos
switch(config-scheduler)# job name offpeakzoning
switch(config-scheduler)# time monthly 28:23:00
switch(config-scheduler)# copy running-config startup-config
switch(config-scheduler)#
```

# Clearing the Scheduler Log File

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **clear scheduler logfile** | Clears the scheduler log file. |

### Example

This example shows how to clear the scheduler log file:

```
switch# configure terminal
switch(config)# clear scheduler logfile
```

# Disabling the Scheduler

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **no feature scheduler** | Disables the scheduler. |
| **Step 3** | (Optional) switch(config) # **show scheduler config** | Displays the scheduler configuration. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to disable the scheduler:

```
switch# configure terminal
switch(config) # no feature scheduler
switch(config) # copy running-config startup-config
switch(config) #
```

# Verifying the Scheduler Configuration

Use one of the following commands to verify the configuration:

*Table 16: Scheduler Show Commands*

| Command | Purpose |
|---|---|
| **show scheduler config** | Displays the scheduler configuration. |
| **show scheduler job** [**name** *name*] | Displays the jobs configured. |
| **show scheduler logfile** | Displays the contents of the scheduler log file. |
| **show scheduler schedule** [**name** *name*] | 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 (the filename is created 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-schedule)# job name backup-cfg
switch(config-schedule)# time daily 1:00
switch(config-schedule)# 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 2009
Last Completion Time: Fri Jan 2 1:00:01 2009
Execution count     : 2
----------------------------------------------
     Job Name            Last Execution Status
----------------------------------------------
back-cfg                          Success (0)
switch(config)#
```

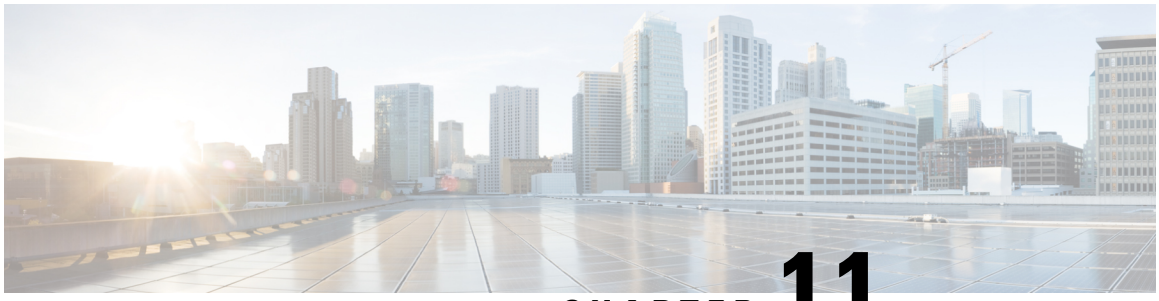
# Displaying the Results of Running Scheduler Jobs

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 2009
-------------------------------- Job Output --------------------------------
`cli var name timestamp 2009-01-01-01.00.00`
`copy running-config bootflash:/$(HOSTNAME)-cfg.$(timestamp)`
`copy bootflash:/switch-cfg.2009-01-01-01.00.00 tftp://1.2.3.4/ vrf management `
copy: cannot access file '/bootflash/switch-cfg.2009-01-01-01.00.00'
==============================================================================
Job Name       : back-cfg                        Job Status: Success (0)
Schedule Name  : daily                           User Name : admin
Completion time: Fri Jan 2  1:00:01 2009
-------------------------------- Job Output --------------------------------
`cli var name timestamp 2009-01-02-01.00.00`
`copy running-config bootflash:/switch-cfg.2009-01-02-01.00.00`
`copy bootflash:/switch-cfg.2009--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#
```

# Standards for the Scheduler

No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.



CHAPTER 11

# Configuring SNMP

This chapter contains the following sections:


- Information About SNMP, on page 127
- Guidelines and Limitations for SNMP, on page 131
- Default SNMP Settings, on page 131
- Configuring SNMP, on page 132
- Configuring the SNMP Local Engine ID, on page 143
- Disabling SNMP, on page 144
- Verifying the SNMP Configuration, on page 144


# Information About 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.

## 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

**Note** Cisco NX-OS does not support SNMP sets for Ethernet MIBs.

The Cisco Nexus device supports SNMPv1, SNMPv2c, and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security.

SNMP is defined in RFC 3410 (http://tools.ietf.org/html/rfc3410), RFC 3411 (http://tools.ietf.org/html/rfc3411), RFC 3412 (http://tools.ietf.org/html/rfc3412), RFC 3413 (http://tools.ietf.org/html/rfc3413), RFC 3414 (http://tools.ietf.org/html/rfc3414), RFC 3415 (http://tools.ietf.org/html/rfc3415), RFC 3416 (http://tools.ietf.org/html/rfc3416), RFC 3417 (http://tools.ietf.org/html/rfc3417), RFC 3418 (http://tools.ietf.org/html/rfc3418), and RFC 3584 (http://tools.ietf.org/html/rfc3584).

# 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 switch 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 Cisco Nexus 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.

# SNMPv3

SNMPv3 provides secure access to devices by a combination of authenticating and encrypting frames over the network. The security features provided in SNMPv3 are the following:

- Message integrity—Ensures that a packet has not been tampered with in-transit.
- Authentication—Determines the message is from a valid source.
- Encryption—Scrambles the packet contents to prevent it from being seen by unauthorized sources.

SNMPv3 provides for both security models and security levels. 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.

## Security Models and Levels for SNMPv1, v2, and 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.

*Table 17: 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 or HMAC-SHA | 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 or HMAC-SHA | 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 Chaning (CBC) DES (DES-56) standard. |

## User-Based Security Model

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—Confirms that the claimed identity of the user who received the data was originated.
- 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 two authentication protocols for SNMPv3:

- HMAC-MD5-96 authentication protocol
- HMAC-SHA-96 authentication protocol

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 indicates 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 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 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 user configuration in the following ways:

- The **auth** passphrase specified in the **snmp-server user** command becomes the password for the CLI user.
- The password specified in the **username** command becomes the **auth** and **priv** 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 passphrase/password in localized key/encrypted format, Cisco NX-OS does not synchronize the user information (passwords, rules, etc.).

## Group-Based SNMP Access

**Note** Because a group is a standard SNMP term used industry-wide, roles are referred to 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 three accesses: read access, write access, and notification access. Each access can be enabled or disabled within each group.

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.

# Guidelines and Limitations for SNMP

SNMP has the following configuration guidelines and limitations:

- 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.
- Cisco NX-OS supports read-only access to Ethernet MIBs. For more information, see the Cisco NX-OS MIB support list at the following URL http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml.
- Cisco NX-OS does not support the SNMPv3 noAuthNoPriv security level.
- For a nondisruptive downgrade path from Cisco NX-OS Release 7.0(3)I6(1) 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.
- Cisco Nexus 3000 series switches support upto 10000 flash files for request.

# Default SNMP Settings

*Table 18: Default SNMP Parameters*

| Parameters | Default |
|---|---|
| license notifications | Enabled |
| linkUp/Down notification type | ietf-extended |

# Configuring SNMP

## Configuring the SNMP Source Interface

You can configure SNMP to use a specific interface.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **snmp-server source-interface** {**inform** \| **trap**} *type slot/port* | Configures the source interface for all SNMP packets. The following list contains the valid values for *interface*.<br>• ethernet<br>• loopback<br>• mgmt<br>• port-channel<br>• vlan |
| **Step 3** | switch(config)# **show snmp source-interface** | Displays the configured SNMP source interface. |

### Example

This example shows how to configure the SNMP source interface:

```
switch(config)# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server source-interface inform ethernet 1/10
switch(config)# snmp-server source-interface trap ethernet 1/10
switch(config)# show snmp source-interface
-----------------------------------------------------------------------
Notification                    source-interface
-----------------------------------------------------------------------
trap                              Ethernet1/10
inform                              Ethernet1/10
-----------------------------------------------------------------------
```

# Configuring SNMP Users

**Note** The commands used to configure SNMP users in Cisco NX-OS are different from those used to configure users in Cisco IOS.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | switch(config)# **snmp-server user** *name* [**auth** {**md5** \| **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID** *id*] [**localizedkey**]]<br>**Example:**<br>`switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh` | Configures an SNMP user with authentication and privacy parameters.<br>The passphrase can be any case-sensitive, alphanumeric string up to 64 characters.<br>If you use the **localizedkey** keyword, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters.<br>The engineID format is a 12-digit, colon-separated decimal number. |
| **Step 3** | (Optional) switch# **show snmp user**<br>**Example:**<br>`switch(config) # show snmp user` | Displays information about one or more SNMP users. |
| **Step 4** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure an SNMP user:

```
switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh
```

# Enforcing 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 that uses a security level parameter of either **noAuthNoPriv** or **authNoPriv**.

Use the following command in global configuration mode to enforce SNMP message encryption for a specific user:

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server user** *name* **enforcePriv** | Enforces SNMP message encryption for this user. |

Use the following command in global configuration mode to enforce SNMP message encryption for all users:

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server globalEnforcePriv** | Enforces SNMP message encryption for all users. |

# Assigning SNMPv3 Users to Multiple Roles

After you configure an SNMP user, you can assign multiple roles for the user.

**Note** Only users who belong to a network-admin role can assign roles to other users.

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server user** *name group* | Associates this SNMP user with the configured user role. |

# Creating SNMP Communities

You can create SNMP communities for SNMPv1 or SNMPv2c.

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server community** *name group* {**ro** \| **rw**} | Creates an SNMP community string. |

# Filtering SNMP Requests

You can assign an access list (ACL) to a community to filter incoming 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)

The ACL applies to both IPv4 and IPv6 over UDP and TCP. After creating the ACL, assign the ACL to the SNMP community.

**Tip** For more information about creating ACLs, see the NX-OS security configuration guide for the Cisco Nexus Series software that you are using.

Use the following command in global configuration mode to assign an ACL to a community to filter SNMP requests:

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server community** *community name* **use-acl** *acl-name*<br>**Example:**<br>`switch(config)# snmp-server community public use-acl my_acl_for_public` | Assigns an IPv4 or IPv6 ACL to an SNMP community to filter SNMP requests. |

# Configuring SNMP Notification Receivers

You can configure Cisco NX-OS to generate SNMP notifications to multiple host receivers.

You can configure a host receiver for SNMPv1 traps in a global configuration mode.

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server host** *ip-address* **traps version 1** *community* [**udp_port** *number*] | Configures a host receiver for SNMPv1 traps. The *ip-address* can be an IPv4 or IPv6 address. The community can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. |

You can configure a host receiver for SNMPv2c traps or informs in a global configuration mode.

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server host** *ip-address* {**traps** \| **informs**} **version 2c** *community* [**udp_port** *number*] | Configures a host receiver for SNMPv2c traps or informs. The *ip-address* can be an IPv4 or IPv6 address. The community can be any alphanumeric string up to 255 characters. The UDP port number range is from 0 to 65535. |

You can configure a host receiver for SNMPv3 traps or informs in a global configuration mode.

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server host** *ip-address* {**traps** \| **informs**} **version 3** {**auth** \| **noauth** \| **priv**} *username* [**udp_port** *number*] | Configures a host receiver for SNMPv2c traps or informs. The *ip-address* can be an IPv4 or IPv6 address. The username 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 engineID of the Cisco Nexus device to authenticate and decrypt the SNMPv3 messages.

The following example shows how to configure a host receiver for an SNMPv1 trap:

```
switch(config)# snmp-server host 192.0.2.1 traps version 1 public
```

The following example shows how to configure a host receiver for an SNMPv2 inform:

```
switch(config)# snmp-server host 192.0.2.1 informs version 2c public
```

The following example shows how to configure a host receiver for an SNMPv3 inform:

```
switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS
```

# Configuring SNMP Notification Receivers with VRFs

You can configure Cisco NX-OS to use a configured VRF to reach the host receiver. 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.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch# **snmp-server host** *ip-address* **use-vrf** *vrf_name* [**udp_port** *number*] | 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 255 characters. The UDP port number range is from 0 to 65535. This command adds an entry into thc ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure the SNMP server host with IP address 192.0.2.1 to use the VRF named "Blue:"

```
switch# configuration terminal
switch(config)# snmp-server host 192.0.2.1 use-vrf Blue
switch(config)# copy running-config startup-config
```

# Filtering SNMP Notifications Based on a VRF

You can configure Cisco NX-OS filter notifications based on the VRF in which the notification occurred.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **snmp-server host** *ip-address* **filter-vrf** *vrf_name* [**udp_port** *number*] | Filters notifications to the notification host receiver based on the configured VRF. The IP address 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.<br>This command adds an entry into thc ExtSnmpTargetVrfTable of the CISCO-SNMP-TARGET-EXT-MB. |
| **Step 3** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure filtering of SNMP notifications based on a VRF:

```
switch# configuration terminal
switch(config)# snmp-server host 192.0.2.1 filter-vrf Red
switch(config)# copy running-config startup-config
```

# Configuring SNMP for Inband Access

You can configure SNMP for inband access using the following:

- Using SNMP v2 without context—You can use a community that is mapped to a context. In this case, the SNMP client does not need to know about the context.
- Using SNMP v2 with context—The SNMP client needs to specify the context by specifying a community; for example, <community>@<context>.
- Using SNMP v3—You can specify the context.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configuration terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **snmp-server context** *context-name* **vrf** *vrf-name* | Maps an SNMP context to the management VRF or default VRF. Custom VRFs are not supported.<br>The names can be any alphanumeric string up to 32 characters. |
| **Step 3** | switch(config)# **snmp-server community** *community-name* **group** *group-name* | Maps an SNMPv2c community to an SNMP context and identifies the group to which the community belongs. The names can be any alphanumeric string up to 32 characters. |
| **Step 4** | switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name* | Maps an SNMPv2c community to an SNMP context. The names can be any alphanumeric string up to 32 characters. |

### Example

The following SNMPv2 example shows how to map a community named snmpdefault to a context:

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)# snmp-server community snmpdefault group network-admin
switch(config)# snmp-server mib community-map snmpdefault context def
switch(config)#
```

The following SNMPv2 example shows how to configure and inband access to the community comm which is not mapped:

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)# snmp-server community comm group network-admin
switch(config)#
```

The following SNMPv3 example shows how to use a v3 username and password:

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)#
```

# Enabling SNMP Notifications

You can enable or disable notifications. If you do not specify a notification name, Cisco NX-OS enables all notifications.

**Note** The **snmp-server enable traps** CLI command enables both traps and informs, depending on the configured notification host receivers.

The following table lists the CLI commands that enable the notifications for Cisco NX-OS MIBs.

***Table 19: Enabling SNMP Notifications***

| MIB | Related Commands |
|---|---|
| All notifications | **snmp-server enable traps** |
| CISCO-ERR-DISABLE-MIB | **snmp-server enable traps show interface status** |
| Q-BRIDGE-MIB | **snmp-server enable traps show mac address-table** |
| CISCO-SWITCH-QOS-MIB | **snmp-server enable traps show hardware internal buffer info pkt-stats** |
| BRIDGE-MIB | **snmp-server enable traps bridge newroot**<br>**snmp-server enable traps bridge topologychange** |
| CISCO-AAA-SERVER-MIB | **snmp-server enable traps aaa** |
| ENITY-MIB,<br>CISCO-ENTITY-FRU-CONTROL-MIB,<br>CISCO-ENTITY-SENSOR-MIB | **snmp-server enable traps entity**<br>**snmp-server enable traps entity fru** |
| CISCO-LICENSE-MGR-MIB | **snmp-server enable traps license** |
| IF-MIB | **snmp-server enable traps link** |
| CISCO-PSM-MIB | **snmp-server enable traps port-security** |
| SNMPv2-MIB | **snmp-server enable traps snmp**<br>**snmp-server enable traps snmp authentication** |
| CISCO-FCC-MIB | **snmp-server enable traps fcc** |
| CISCO-DM-MIB | **snmp-server enable traps fcdomain** |
| CISCO-NS-MIB | **snmp-server enable traps fcns** |

| MIB | Related Commands |
|---|---|
| CISCO-FCS-MIB | **snmp-server enable traps fcs discovery-complete**<br>**snmp-server enable traps fcs request-reject** |
| CISCO-FDMI-MIB | **snmp-server enable traps fdmi** |
| CISCO-FSPF-MIB | **snmp-server enable traps fspf** |
| CISCO-PSM-MIB | **snmp-server enable traps port-security** |
| CISCO-RSCN-MIB | **snmp-server enable traps rscn**<br>**snmp-server enable traps rscn els**<br>**snmp-server enable traps rscn ils** |
| CISCO-ZS-MIB | **snmp-server enable traps zone**<br>**snmp-server enable traps zone default-zone-behavior-change**<br>**snmp-server enable traps zone enhanced-zone-db-change**<br>**snmp-server enable traps zone merge-failure**<br>**snmp-server enable traps zone merge-success**<br>**snmp-server enable traps zone request-reject**<br>**snmp-server enable traps zone unsupp-mem** |
| CISCO-CONFIG-MAN-MIB<br>**Note**<br>Supports no MIB objects except the following notification: ccmCLIRunningConfigChanged | **snmp-server enable traps config** |

**Note** The license notifications are enabled by default.

To enable the specified notification in the global configuration mode, perform one of the following tasks:

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server enable traps** | Enables all SNMP notifications. |
| switch(config)# **snmp-server enable traps aaa** [**server-state-change**] | Enables the AAA SNMP notifications. |
| switch(config)# **snmp-server enable traps entity** [**fru**] | Enables the ENTITY-MIB SNMP notifications. |
| switch(config)# **snmp-server enable traps license** | Enables the license SNMP notification. |
| switch(config)# **snmp-server enable traps port-security** | Enables the port security SNMP notifications. |

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server enable traps snmp** [**authentication**] | Enables the SNMP agent notifications. |

# Configuring Link Notifications

You can configure which linkUp/linkDown notifications to enable on a device. You can enable the following types of linkUp/linkDown notifications:

- cieLinkDown—Enables the Cisco extended link state down notification.
- cieLinkUp—Enables the Cisco extended link state up notification.
- cisco-xcvr-mon-status-chg—Enables the Cisco interface transceiver monitor status change notification.
- delayed-link-state-change—Enables the delayed link state change.
- extended-linkUp—Enables the Internet Engineering Task Force (IETF) extended link state up notification.
- extended-linkDown—Enables the IETF extended link state down notification.
- linkDown—Enables the IETF Link state down notification.
- linkUp—Enables the IETF Link state up notification.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **snmp-server enable traps link** [**cieLinkDown** \| **cieLinkUp** \| **cisco-xcvr-mon-status-chg** \| **delayed-link-state-change**] \| **extended-linkUp** \| **extended-linkDown** \| **linkDown** \| **linkUp**]<br>**Example:**<br>`switch(config)# snmp-server enable traps link cieLinkDown` | Enables the link SNMP notifications. |

# Disabling Link Notifications on an Interface

You can disable linkUp and linkDown notifications on an individual interface. You can use these limit notifications on a flapping interface (an interface that transitions between up and down repeatedly).

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **interface** *type slot/port* | Specifies the interface to be changed. |
| **Step 3** | switch(config -if)# **no snmp trap link-status** | Disables SNMP link-state traps for the interface. This feature is enabled by default. |

# Enabling One-Time Authentication for SNMP over TCP

You can enable a one-time authentication for SNMP over a TCP session.

| Command | Purpose |
|---|---|
| switch(config)# **snmp-server tcp-session** [**auth**] | Enables a one-time authentication for SNMP over a TCP session. This feature is disabled by default. |

# Assigning SNMP Switch Contact and Location Information

You can assign the switch contact information, which is limited to 32 characters (without spaces), and the switch location.

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configuration terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **snmp-server contact** *name* | Configures sysContact, the SNMP contact name. |
| **Step 3** | switch(config)# **snmp-server location** *name* | Configures sysLocation, the SNMP location. |
| **Step 4** | (Optional) switch# **show snmp** | Displays information about one or more destination profiles. |
| **Step 5** | (Optional) switch# **copy running-config startup-config** | Saves this configuration change. |

# Configuring the 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.

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configuration terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*] | Maps an SNMP context to a protocol instance, VRF, or topology. The names can be any alphanumeric string up to 32 characters. |
| **Step 3** | switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name* | Maps an SNMPv2c community to an SNMP context. The names can be any alphanumeric string up to 32 characters. |
| **Step 4** | (Optional) switch(config)# **no snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*] | Deletes the mapping between an SNMP context and a protocol instance, VRF, or topology. The names can be any alphanumeric string up to 32 characters.<br>**Note**<br>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. |

# Configuring the 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**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **snmp-server engineID local** *engineid-string*<br>**Example:** | Changes the SNMP engineID of the local device. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch(config)# snmp-server engineID local AA:BB:CC:1A:2C:10` | The local engine ID should be configured as a list of colon-specified hexadecimal octets, where there are even number of hexadecimal characters that range from 10 to 64 and every two hexadecimal characters are separated by a colon. For example, i80:00:02:b8:04:61:62:63. |
| **Step 3** | **show snmp engineID**<br>**Example:**<br>`switch(config)# show snmp engineID` | Displays the identification of the configured SNMP engine. |
| **Step 4** | **[no] snmp-server engineID local** *engineid-string*<br>**Example:**<br>`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**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

# Disabling SNMP

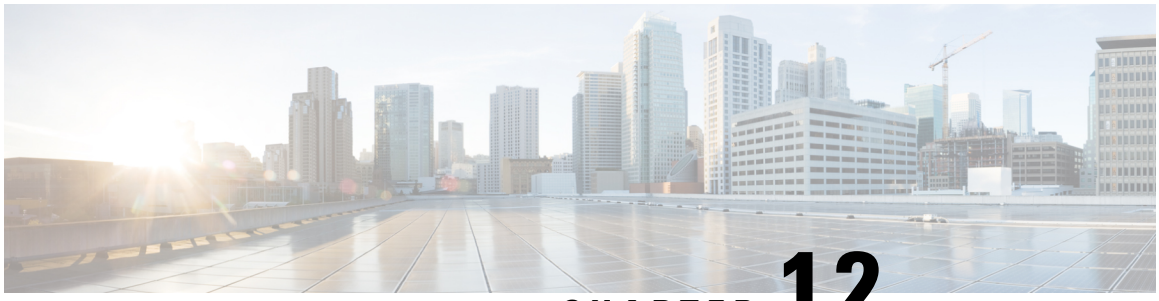
### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | switch(config) # **no snmp-server protocol enable**<br>**Example:**<br>`no snmp-server protocol enable` | Disables SNMP.<br>SNMP is disabled by default. |

# Verifying the SNMP Configuration

To display SNMP configuration information, perform one of the following tasks:

| Command | Purpose |
|---|---|
| **show snmp** | Displays the SNMP status. |
| **show snmp community** | Displays the SNMP community strings. |
| **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. |
| **show snmp engineID** | Displays the SNMP engineID. |
| **show snmp group** | Displays SNMP roles. |
| **show snmp sessions** | Displays SNMP sessions. |
| **show snmp context** | Displays the SNMP context mapping. |
| **show snmp host** | Displays information about configured SNMP hosts. |
| **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. |



CHAPTER 12

# Using the PCAP SNMP Parser

This chapter contains the following sections:


- Using the PCAP SNMP Parser, on page 147


# Using the PCAP SNMP Parser

The PCAP SNMP parser is a tool to analyze SNMP packets captured in .pcap format. It runs on the switch and generates a statistics report for all of the SNMP get, getnext, getbulk, set, trap, and response requests sent to the switch.

To use the PCAP SNMP parser, use one of the following commands:

- **debug packet-analysis snmp** [**mgmt0** | **inband**] **duration** *seconds* [*output-file*] [**keep-pcap**]—Captures packets for a specified number of seconds using Tshark, saves them in a temporary .pcap file, and then analyzes them based on this .pcap file.

  The results are saved in the output file or printed to the console, if the output file is not specified. The temporary .pcap file will be deleted by default, unless you use the **keep-pcap** option. Packet capture can be performed on the management interface (mgmt0), which is the default, or the inband interface.

  **Examples:**

```
switch# debug packet-analysis snmp duration 100

switch# debug packet-analysis snmp duration 100 bootflash:snmp_stats.log

switch# debug packet-analysis snmp duration 100 bootflash:snmp_stats.log keep-pcap

switch# debug packet-analysis snmp inband duration 100

switch# debug packet-analysis snmp inband duration 100 bootflash:snmp_stats.log

switch# debug packet-analysis snmp inband duration 100 bootflash:snmp_stats.log keep-pcap
```

- **debug packet-analysis snmp** *input-pcap-file* [*output-file*]—Analyzes the captured packets on an existing .pcap file.

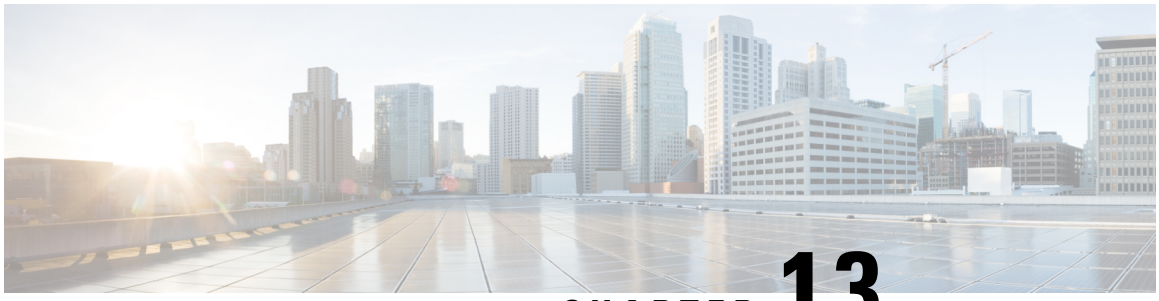
  **Examples:**

```
switch# debug packet-analysis snmp bootflash:snmp.pcap
```

```
switch# debug packet-analysis snmp bootflash:snmp.pcap bootflash:snmp_stats.log
```

The following example shows a sample statistics report for the **debug packet-analysis snmp** [**mgmt0** | **inband**] **duration** command:

```
switch# debug packet-analysis snmp duration 10
Capturing on eth0
36
wireshark-cisco-mtc-dissector: ethertype=0xde09, devicetype=0x0
wireshark-broadcom-rcpu-dissector: ethertype=0xde08, devicetype=0x0

Started analyzing. It may take several minutes, please wait!

Statistics Report
----------------------------------------
SNMP Packet Capture Duration: 0 seconds
Total Hosts: 1
Total Requests: 18
Total Responses: 18
Total GET: 0
Total GETNEXT: 0
Total WALK: 1 (NEXT: 18)
Total GETBULK: 0
Total BULKWALK: 0 (BULK: 0)
Total SET: 0
Total TRAP: 0
Total INFORM: 0

Hosts         GET  GETNEXT  WALK(NEXT) GETBULK  BULKWALK(BULK) SET  TRAP  INFORM  RESPONSE
-------------------------------------------------------------------------------------------
10.22.27.244   0     0        1(18)      0         0(0)          0    0      0       18

Sessions
--------
1

MIB Objects GET  GETNEXT  WALK(NEXT) GETBULK(Non_rep/Max_rep) BULKWALK(BULK, Non_rep/Max_rep)
-------------------------------------------------------------------------------------------
ifName      0       0        1(18)      0                         0


SET     Hosts
--------------------
0       10.22.27.244
```



CHAPTER 13

# Configuring RMON

This chapter contains the following sections:


- Information About RMON, on page 149
- Configuration Guidelines and Limitations for RMON, on page 150
- Verifying the RMON Configuration, on page 150
- Default RMON Settings, on page 151
- Configuring RMON Alarms, on page 151
- Configuring RMON Events, on page 152


# Information About RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. The Cisco NX-OS supports RMON alarms, events, and logs to monitor Cisco Nexus device.

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 disabled by default and no events or alarms are configured in Cisco Nexus devices. You can configure your RMON alarms and events 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, 1.3.6.1.2.1.2.2.1.17 represents ifOutOctets.17).

When you create an alarm, you specify the following parameters:

- MIB object to monitor
- Sampling interval—The interval that the Cisco Nexus 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 Cisco Nexus device triggers a rising alarm or resets a falling alarm.
- Falling threshold—The value at which theCisco Nexus device triggers a falling alarm or resets a rising alarm.
- Events—The action that the Cisco Nexus 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 does 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.

# Configuration 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 only configure an RMON alarm on a MIB object that resolves to an integer.

# Verifying the RMON Configuration

Use the following commands to verify the RMON configuration information:

| Command | Purpose |
|---|---|
| **show rmon alarms** | Displays information about RMON alarms. |

| Command | Purpose |
|---|---|
| **show rmon events** | Displays information about RMON events. |
| **show rmon hcalarms** | Displays information about RMON hcalarms. |
| **show rmon logs** | Displays information about RMON logs. |

# Default RMON Settings

The following table lists the default settings for RMON parameters.

***Table 20: Default RMON Parameters***

| Parameters | Default |
|---|---|
| Alarms | None configured. |
| Events | None configured. |

# Configuring RMON Alarms

You can configure RMON alarms on any integer-based SNMP MIB object.

You can optionally specify the following parameters:

- The eventnumber 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

Ensure you have configured an SNMP user and enabled SNMP notifications.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **rmon alarm** *index mib-object sample-interval* {**absolute** \| **delta**} **rising-threshold** *value* [*event-index*] **falling-threshold** *value* [*event-index*] [**owner** *name*] | Creates an RMON alarm. The value range is from –2147483647 to 2147483647. The owner name can be any alphanumeric string. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config)# **rmon hcalarm** *index mib-object sample-interval* {**absolute** \| **delta**} **rising-threshold-high** *value* **rising-threshold-low** *value* [*event-index*] **falling-threshold-high** *value* **falling-threshold-low** *value* [*event-index*] [**owner** *name*] [**storagetype** *type*] | Creates an RMON high-capacity alarm. The value range is from –2147483647 to 2147483647. The owner name can be any alphanumeric string.<br>The storage type range is from 1 to 5. |
| **Step 4** | (Optional) switch# **show rmon** {**alarms** \| **hcalarms**} | Displays information about RMON alarms or high-capacity alarms. |
| **Step 5** | (Optional) switch# **copy running-config startup-config** | Saves this configuration change. |

### Example

The following example shows how to configure RMON alarms:

```
switch# configure terminal
switch(config)# rmon alarm 1 1.3.6.1.2.1.2.2.1.17.83886080 5 delta rising-threshold 5 1
falling-threshold 0 owner test
switch(config)# exit
switch# show rmon alarms
Alarm 1 is active, owned by test
 Monitors 1.3.6.1.2.1.2.2.1.17.83886080 every 5 second(s)
 Taking delta samples, last value was 0
 Rising threshold is 5, assigned to event 1
 Falling threshold is 0, assigned to event 0
 On startup enable rising or falling alarm
```

# Configuring RMON Events

You can configure RMON events to associate with RMON alarms. You can reuse the same event with multiple RMON alarms.

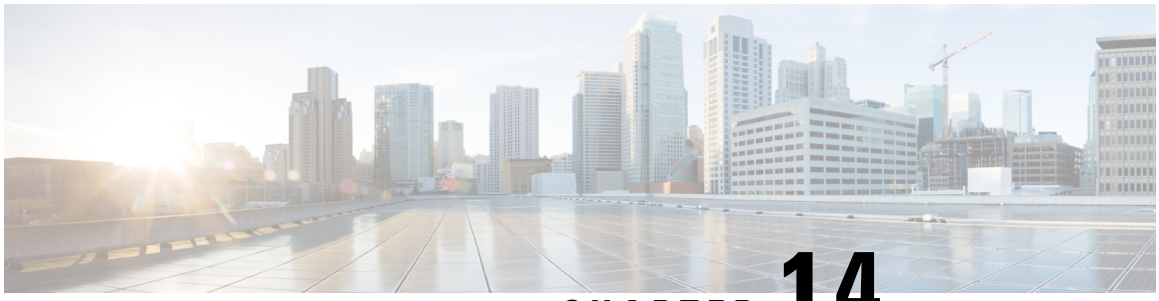
Ensure you have configured an SNMP user and enabled SNMP notifications.

### Before you begin

Ensure that you have configured an SNMP user and enabled SNMP notifications.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **rmon event** *index* [**description** *string*] [**log**] [**trap**] [**owner** *name*] | Configures an RMON event. The description string and owner name can be any alphanumeric string. |
| **Step 3** | (Optional) switch(config)# **show rmon** {**alarms** \| **hcalarms**} | Displays information about RMON alarms or high-capacity alarms. |
| **Step 4** | (Optional) switch# **copy running-config startup-config** | Saves this configuration change. |



CHAPTER 14

# Configuring Online Diagnostics

This chapter contains the following sections:


- Information About Online Diagnostics, on page 155
- Guidelines and Limitations for Online Diagnostics, on page 157
- Configuring Online Diagnostics, on page 158
- Verifying the Online Diagnostics Configuration, on page 159
- Default Settings for Online Diagnostics, on page 159
- Parity Error Diagnostics, on page 159


# Information About Online Diagnostics

Online diagnostics provide verification of hardware components during switch bootup or reset, and they monitor the health of the hardware during normal switch operation.

Cisco Nexus Series switches support bootup diagnostics and runtime diagnostics. Bootup diagnostics include disruptive tests and nondisruptive tests that run during system bootup and system reset.

Runtime diagnostics (also known as health monitoring diagnostics) include nondisruptive tests that run in the background during normal operation of the switch.

## Bootup Diagnostics

Bootup diagnostics detect faulty hardware before bringing the switch online. Bootup diagnostics also check the data path and control path connectivity between the supervisor and the ASICs. The following table describes the diagnostics that are run only during switch bootup or reset.

***Table 21: Bootup Diagnostics***

| Diagnostic | Description |
|---|---|
| PCIe | Tests PCI express (PCIe) access. |
| NVRAM | Verifies the integrity of the NVRAM. |
| In band port | Tests connectivity of the inband port to the supervisor. |
| Management port | Tests the management port. |

| Diagnostic | Description |
|---|---|
| Memory | Verifies the integrity of the DRAM. |

Bootup diagnostics also include a set of tests that are common with health monitoring diagnostics.

Bootup diagnostics log any failures to the onboard failure logging (OBFL) system. Failures also trigger an LED display to indicate diagnostic test states (on, off, pass, or fail).

You can configure Cisco Nexus device to either bypass the bootup diagnostics or run the complete set of bootup diagnostics.

# Health Monitoring Diagnostics

Health monitoring diagnostics provide information about the health of the switch. They detect runtime hardware errors, memory errors, software faults, and resource exhaustion.

Health monitoring diagnostics are nondisruptive and run in the background to ensure the health of a switch that is processing live network traffic.

The following table describes the health monitoring diagnostics for the switch.

***Table 22: Health Monitoring Diagnostics Tests***

| Diagnostic | Description |
|---|---|
| LED | Monitors port and system status LEDs. |
| Power Supply | Monitors the power supply health state. |
| Temperature Sensor | Monitors temperature sensor readings. |
| Test Fan | Monitors the fan speed and fan control. |

**Note** When the switch reaches the intake temperature threshold and does not go within the limits in 120 seconds, the switch will power off and the power supplies will have to be re-seated to recover the switch

The following table describes the health monitoring diagnostics that also run during system boot or system reset.

***Table 23: Health Monitoring and Bootup Diagnostics Tests***

| Diagnostic | Description |
|---|---|
| SPROM | Verifies the integrity of backplane and supervisor SPROMs. |
| Fabric engine | Tests the switch fabric ASICs. |
| Fabric port | Tests the ports on the switch fabric ASIC. |
| Forwarding engine | Tests the forwarding engine ASICs. |
| Forwarding engine port | Tests the ports on the forwarding engine ASICs. |

| Diagnostic | Description |
|---|---|
| Front port | Tests the components (such as PHY and MAC) on the front ports. |

**Note** When the switch exceeds the internal temperature threshold of 70 degrees Celsius and does not decrease below the threshold limit within 120 seconds, the switch powers off and the switch must be properly power-cycled in order to recover the switch.

# Expansion Module Diagnostics

During the switch bootup or reset, the bootup diagnostics include tests for the in-service expansion modules in the switch.

When you insert an expansion module into a running switch, a set of diagnostics tests are run. The following table describes the bootup diagnostics for an expansion module. These tests are common with the bootup diagnostics. If the bootup diagnostics fail, the expansion module is not placed into service.

***Table 24: Expansion Module Bootup and Health Monitoring Diagnostics***

| Diagnostic | Description |
|---|---|
| SPROM | Verifies the integrity of backplane and supervisor SPROMs. |
| Fabric engine | Tests the switch fabric ASICs. |
| Fabric port | Tests the ports on the switch fabric ASIC. |
| Forwarding engine | Tests the forwarding engine ASICs. |
| Forwarding engine port | Tests the ports on the forwarding engine ASICs. |
| Front port | Tests the components (such as PHY and MAC) on the front ports. |

Health monitoring diagnostics are run on in-service expansion modules. The following table describes the additional tests that are specific to health monitoring diagnostics for expansion modules.

***Table 25: Expansion Module Health Monitoring Diagnostics***

| Diagnostic | Description |
|---|---|
| LED | Monitors port and system status LEDs. |
| Temperature Sensor | Monitors temperature sensor readings. |

# Guidelines and Limitations for Online Diagnostics

Online diagnostics has the following configuration guidelines and limitations:

- You cannot run disruptive online diagnostic tests on demand.

- The BootupPortLoopback test is not supported.
- Interface Rx and Tx packet counters are incremented (approximately four packets every 15 minutes) for ports in the shutdown state.
- On admin down ports, the unicast packet Rx and Tx counters are incremented for GOLD loopback packets. The PortLoopback test is on demand for releases prior to Cisco NX-OS 7.0(3)I1(2), so the packet counter is incremented only when you run the test on admin down ports. Starting with Cisco NX-OS Release 7.0(3)I1(2), 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.

# Configuring Online Diagnostics

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 boot up 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** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **diagnostic bootup level** [**complete** \| **bypass**] | Configures the bootup diagnostic level to trigger diagnostics when the device boots, as follows:<br>• **complete**—Performs all bootup diagnostics. This is the default value.<br>• **bypass**—Does not perform any bootup diagnostics. |
| **Step 3** | (Optional) switch# **show diagnostic bootup level** | Displays the bootup diagnostic level (bypass or complete) that is currently in place on the switch. |

### Example

The following example shows how to configure the bootup diagnostics level to trigger the complete diagnostics:

```
switch# configure terminal
switch(config)# diagnostic bootup level complete
```

# Verifying the Online Diagnostics Configuration

Use the following commands to verify online diagnostics configuration information:

| Command | Purpose |
|---|---|
| **show diagnostic bootup level** | Displays the bootup diagnostics level. |
| **show diagnostic result module** *slot* | Displays the results of the diagnostics tests. |

# Default Settings for Online Diagnostics

The following table lists the default settings for online diagnostics parameters.

***Table 26: Default Online Diagnostics Parameters***

| Parameters | Default |
|---|---|
| Bootup diagnostics level | complete |

# Parity Error Diagnostics

## Clearing Parity Errors

You can clear a corresponding Layer 2 or Layer 3 table entry (with 0s) when a parity error is detected by using the **hardware profile parity-error {l2-table | l3-table} clear** command. This command is effective when it is present in the running configuration and the system is booting up. In addition, the command must be enabled and after the configuration is saved, the system should be rebooted for the command to take effect.

**Important** This command is not supported on Cisco NX-OS Release 6.0(2)U2(1) and higher versions.

The following guidelines apply:

- When the command is used for an l2_entry table, the cleared entry should be relearned due to the traffic pattern.
- When the command is used for an l3_entry_only (host) table, the cleared entry is not be relearned.

The command is useful in the following customer configurations:

- L2_Entry table, with no static L2_entry table entries

  If the L2_Entry table entry is cleared, the entry should be dynamically learned through the traffic pattern. It should not be learned through IGMP or multicast.

- L3_Entry_only (host) table

Customers should not use the host table. The **hardware profile unicast enable-host-ecmp** command should be enabled. In this case, the customer node does not have any valid entries in the L3_Entry_only table, so clearing the L3_Entry_only entry table should not have any impact.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **hardware profile parity-error l2-table clear** | Clears parity error entries in a Layer 2 table. |
| **Step 3** | switch(config)# **hardware profile parity-error l3-table clear** | Clears parity error entries in a Layer 3 table. |

#### Example

This example shows how to clear parity errors in a Layer 2 table:

```
switch# configure terminal
switch(config)# hardware profile parity-error l2-table clear
switch(config)# copy running-config startup-config
switch(config)# reload
```

This example shows how to clear parity errors in a Layer 3 table:

```
switch# configure terminal
switch(config)# hardware profile parity-error l3-table clear
switch(config)# copy running-config startup-config
switch(config)# reload
```

# Soft Error Recovery

Cisco NX-OS Release 6.0(2)U2(1) introduces software error recovery (SER) for soft errors in the internal memory tables of the forwarding engine. This feature is enabled by default.

The forwarding engine internal control tables and packet memories are protected through various mechanisms such as error-correcting code (ECC), parity protection, or software scan based parity check of the tables. Software caches are maintained for most of the hardware tables. Parity and ECC errors are detected when the traffic hits the affected entries. For ternary content addressable memories (TCAMs), an error is detected when the CPU compares the software shadow entries to the hardware entries. When any of these types of errors are detected, an interrupt is generated to report an error for that memory.

The correction mechanism is different for different hardware tables. For hardware tables that have a software shadow, the affected entry is copied from the software cache and the interrupt is cleared. Hardware tables, such as the Layer 3 host lookup table and the ACL TCAM tables, are detected and corrected in this way. For hardware tables that do not have a software shadow, the affected entry is cleared or zeroed out. Hardware tables, such as the hardware-learned Layer 2 entry table, and the counters' memory are detected and corrected in this way.

When a parity error is encountered in the hardware in the forwarding lookup for the packet, the packet is subject to a drop depending on the table encountering the parity error. The recovery time from the parity error

detection to correction, in this case, for an entry can be over 600 microseconds. If the traffic is hitting this entry, there will be traffic loss for this duration.

For TCAM tables that do not have parity protection, a periodic software scan is done for the table entries to detect parity errors. In case of parity error detection, the system copies the affected memory location from the software shadow to correct the error. Software initiated scan is done every 10 seconds with 4,000 entries scanned per interval. There are about 36,000 TCAM entries to be scanned in the forwarding engine. In the worst case scenario, it can take over 90 seconds for parity error detection and correction for these tables, the recovery time is based on the system load.

In case of unrecoverable parity errors, the software generates a syslog event notification as shown in the following example:

```
2013 Nov 14 12:37:32 switch %USER-3-SYSTEM_MSG: bcm_usd_isr_switch_event_cb_log:658: slot_num
 0, event 2, memory error type: Detection(0x1), table name: Ingress ACL result
table(0x830004b5), index: 1790  - bcm_usd
```

# Verifying Memory Table Health

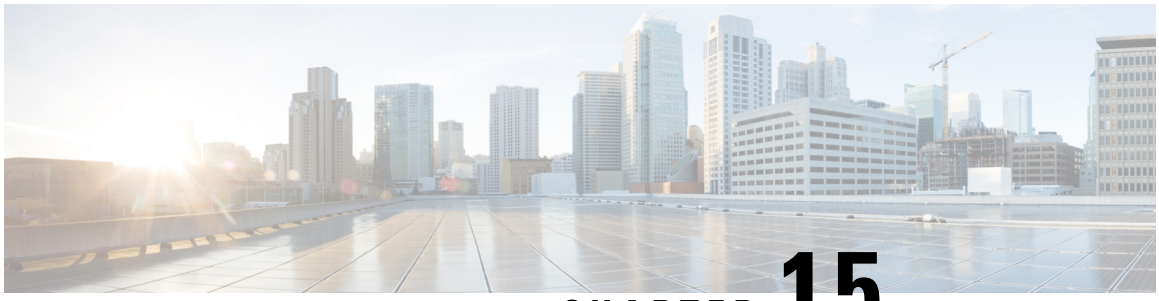
To display a summary of parity error counts encountered in ASIC memory tables, run the following command:

| Command | Purpose |
|---|---|
| **show hardware forwarding memory health summary** | Displays a summary of parity error counts in ASIC memory tables. |

### Example

The following example shows how to display a summary of parity error counts in ASIC memory tables:

```
switch# show hardware forwarding memory health summary
Parity error counters:
Total parity error detections: 7
Total parity error corrections: 7
Total TCAM table parity error detections: 1
Total TCAM table parity error corrections: 1
Total SRAM table parity error detections: 6
Total SRAM table parity error corrections: 6
Parity error summary:
Table ID: L2 table      Detections: 1   Corrections: 1
Table ID: L3 Host table Detections: 1   Corrections: 1
Table ID: L3 LPM table  Detections: 1   Corrections: 1
Table ID: L3 LPM result table   Detections: 1   Corrections: 1
Table ID: Ingress pre-lookup ACL result table   Detections: 1   Corrections: 1
Table ID: Ingress ACL result table      Detections: 1   Corrections: 1
Table ID: Egress ACL result table       Detections: 1   Corrections: 1
```



CHAPTER 15

# Configuring the Embedded Event Manager

This chapter contains the following sections:


- Information About Embedded Event Manager, on page 163
- Configuring Embedded Event Manager, on page 167
- Verifying the Embedded Event Manager Configuration, on page 177
- Configuration Examples for Embedded Event Manager, on page 178
- Additional References, on page 179
- Feature History for EEM, on page 179


## Information About Embedded Event Manager

The ability to detect and handle critical events in the Cisco NX-OS system is important for high availability. The Embedded Event Manager (EEM) provides a central, policy-driven framework to detect and handle events in the system by monitoring events that occur on your device and taking 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 sending an e-mail or disabling an interface, to recover from an event.

**Policies**

An event paired with one or more actions to troubleshoot or recover from the event.

Without EEM, each individual component is responsible for detecting and handling its own events. For example, if a port flaps frequently, the policy of "putting it into errDisable state" is built into ETHPM.

# Embedded Event Manager 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.

For example, you can configure an EEM policy to identify when a card is removed from the device and log the details related to the card removal. By setting up an event statement that tells the system to look for all instances of card removal and an then with an action statement that tells the system to log the details.

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. Once EEM policies are configured, the corresponding actions are triggered. All actions (system or user-configured) for triggered events are tracked and maintained by the system.

### Preconfigured System Policies

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 (__).

Some system policies can be overridden. In these cases, you can configure overrides for either the event or the action. The overrides that you configure take the place of the system policy.

**Note** Override policies must include an event statement. Override policies without event statements override all possible events for the system policy.

To view the preconfigured system polices and determine which polices you can override, use the **show event manager system-policy** command.

### User-Created Policies

User-created policies allow you to customize EEM policies for your network. If a user policy is created for an event, actions in the policy are triggered only after EEM triggers the system policy actions related to the same event.

### Log Files

The log file that contains data that is related to EEM policy matches is maintained in the event_archive_1 log file located in the /log/event_archive_1 directory.

# Event Statements

Any device activity for which some action, such as a workaround or notification, is taken is considered an event by EEM. In many cases, events are related to faults in the device, such as when an interface or a fan malfunctions.

Event statements specify which event or events triggers a policy to run.

**Tip** You can configure EEM to trigger an EEM policy that is based on a combination of events by creating and differentiating multiple EEM events in the policy and then defining a combination of events to trigger a custom action.

EEM defines event filters so that only critical events or multiple occurrences of an event within a specified time period trigger an associated action.

Some commands or internal events trigger other commands internally. These commands are not visible, but will still match the event specification that triggers an action. You cannot prevent these commands from triggering an action, but you can check which event triggered an action.

### Supported Events

EEM supports the following events in event statements:

- Counter events
- Fan absent events
- Fan bad events
- Memory thresholds events
- Events being used in overridden system policies.
- SNMP notification events
- Syslog events
- System manager events
- Temperature events
- Track events

## Action Statements

Action statements describe the action that is triggered by a policy when an event occurs. Each policy can have multiple action statements. If no action is associated with a policy, EEM still observes events but takes no actions.

In order for triggered events to process 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 does not allow the command to execute.

**Note** When configuring action statements within your user policy or overriding policy, it is important that you confirm that action statements do not negate each other or adversely affect the associated system policy.

### Supported Actions

EEM supports the following actions in action statements:

- Execute any CLI commands
- Update a counter
- Reload the device
- Generate a syslog message
- Generate an SNMP notification
- Use the default action for the system policy

# VSH Script Policies

You can write policies in a VSH script, by using a text editor. Policies that are written using a VSH script have an event statement and action statement(s) just as other policies, and these policies can either augment or override system policies.

After you define your VSH script policy, copy it to the device and activate it.

# Licensing Requirements for Embedded Event Manager

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

# Prerequisites for Embedded Event Manager

You must have network-admin privileges to configure EEM.

# Guidelines and Limitations for Embedded Event Manager

When you plan your EEM configuration, consider the following:

- 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.
- If you want 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 command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute.
- An override policy that consists of an event statement and no 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.
- In regular command expressions: all keywords must be expanded, and only the asterisk (*) symbol can be used for replace the arguments.
- 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, snmp, syslog, and track.

- 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.
- EEM event correlation does not override the system default policies.
- Default action execution is not supported for policies that are configured with tagged events.
- If your event specification matches a CLI pattern, you can use SSH-style wild card characters.

  For example, if you want to match all show commands, enter the **show *** command. Entering the **show . *** command does not work.
- If your event specification is a regular expression for a matching syslog message, you can use a proper regular expression.

  For example, if you want to detect ADMIN_DOWN events on any port where a syslog is generated, use **.ADMIN_DOWN.**. Entering the **ADMIN_DOWN** command does not work.
- In the event specification for a syslog, the regex does not match any syslog message that is generated as an action of an EEM policy.
- If an EEM event matches a **show** command in the CLI and you want the output for that **show** command to display on the screen (and to not be blocked by the EEM policy), you must specify the **event-default** command for the first action for the EEM policy.

## Default Settings for Embedded Event Manager

*Table 27: Default EEM Parameters*

| Parameters | Default |
|---|---|
| System Policies | Active |

# Configuring Embedded Event Manager

## Defining an Environment Variable

Defining an environment variable is an optional step but is useful for configuring common values for repeated use in multiple policies.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | **event manager environment** *variable-name variable-value*<br>**Example:**<br>`switch(config) # event manager environment emailto "admin@anyplace.com"` | Creates an environment variable for EEM.<br>The *variable-name* can be any case-sensitive, alphanumeric string up to 29 characters.<br>The *variable-value* can be any quoted case-sensitive, alphanumeric string up to 39 characters. |
| **Step 3** | (Optional) **show event manager environment** {*variable-name* \| **all**}<br>**Example:**<br>`switch(config) # show event manager environment all` | Displays information about the configured environment variables. |
| **Step 4** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### What to do next

Configure a User Policy.

# Defining a User Policy Using the CLI

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **event manager applet** *applet-name*<br>**Example:**<br>`switch(config)# event manager applet monitorShutdown`<br>`switch(config-applet)#` | Registers the applet with EEM and enters applet configuration mode.<br>The applet-name can be any case-sensitive, alphanumeric string up to 29 characters. |
| **Step 3** | (Optional) **description** *policy-description*<br>**Example:**<br>`switch(config-applet)# description "Monitors interface shutdown."` | Configures a descriptive string for the policy.<br>The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | **event** *event-statement*<br>**Example:**<br>`switch(config-applet)# event cli match "shutdown"` | Configures the event statement for the policy. |
| **Step 5** | (Optional) **tag** *tag* {**and** \| **andnot** \| **or**} *tag* [**and** \| **andnot** \| **or** {*tag*}] {**happens** *occurs* **in** *seconds*}<br>**Example:**<br>`switch(config-applet)# tag one or two happens 1 in 10000` | Correlates multiple events in the policy.<br>The range for the *occurs* argument is from 1 to 4294967295.<br>The range for the *seconds* argument is from 0 to 4294967295 seconds. |
| **Step 6** | **action** *number*[.*number2*] *action-statement*<br>**Example:**<br>`switch(config-applet)# action 1.0 cli show interface e 3/1` | Configures an action statement for the policy. Repeat this step for multiple action statements. |
| **Step 7** | (Optional) **show event manager policy-state** *name* [**module** *module-id*]<br>**Example:**<br>`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**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### What to do next

Configure event statements and action statements.

# Configuring Event Statements

Use one of the following commands in EEM configuration mode (config-applet) to configure an event statement:

**Note** When many features are deployed, baseline memory requires to define *minor*, *severe*, and *critical* thresholds. Because the default thresholds are calculated on boot up depending on the DRAM size, its value varies depending on the DRAM size that is used on the platform. You can configure the thresholds using the system memory-thresholds minor percentage severe percentage critical percentage command. For low memory platforms, for example devices with 4GB DRAM, the memory thresholds are set to a higher value to avoid false alarms.

### Before you begin

Define a user policy.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **event cli** [**tag** *tag*] **match** *expression* [**count** *repeats* \| **time** *seconds* <br> **Example:** <br> `switch(config-applet) # event cli match "shutdown"` | Triggers an event if you enter a command that matches the regular expression. <br> The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. <br> The *repeats* range is from 1 to 65000. <br> The *time* range is from 0 to 4294967295, where 0 indicates no time limit. |
| **Step 2** | **event counter** [**tag** *tag*] **name** *counter* **entry-val** *entry* **entry-op** {**eq** \| **ge** \| **gt** \| **le** \| **lt** \| **ne**} {**exit-val** *exit* **exit-op** {**eq** \| **ge** \| **gt** \| **le** \| **lt** \| **ne**} <br> **Example:** <br> `switch(config-applet) # event counter name mycounter entry-val 20 gt` | 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. <br> The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. <br> The *counter* name can be any case-sensitive, alphanumeric string up to 28 characters. <br> The *entry* and *exit* value ranges are from 0 to 2147483647. |
| **Step 3** | **event fanabsent** [**fan** *number*] **time** *seconds* <br> **Example:** <br> `switch(config-applet) # event fanabsent time 300` | Triggers an event if a fan is removed from the device for more than the configured time, in seconds. <br> The *number* range is is from 1 to 1 and is module-dependent. <br> The *seconds* range is from 10 to 64000. |
| **Step 4** | **event fanbad** [**fan** *number*] **time** *seconds* <br> **Example:** <br> `switch(config-applet) # event fanbad time 3000` | Triggers an event if a fan fails for more than the configured time, in seconds. <br> The *number* range is module-dependent. <br> The *seconds* range is from 10 to 64000. |
| **Step 5** | **event memory** {**critical** \| **minor** \| **severe**} <br> **Example:** <br> `switch(config-applet) # event memory critical` | Triggers an event if a memory threshold is crossed. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | **event policy-default count** *repeats* [**time** *seconds*]<br>**Example:**<br>`switch(config-applet) # event policy-default count 3` | Uses the event configured in the system policy. Use this option for overriding policies.<br>The *repeats* range is from 1 to 65000.<br>The *seconds* range is from 0 to 4294967295, where 0 indicates no time limit. |
| **Step 7** | **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*<br>**Example:**<br>`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` | 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.<br>The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy.<br>The *entry* and *exit* value ranges are from 0 to 18446744073709551615.<br>The *time*, in seconds, is from 0 to 2147483647.<br>The *interval*, in seconds, is from 0 to 2147483647. |
| **Step 8** | **event sysmgr memory** [**module** *module-num*] **major** *major-percent* **minor** *minor-percent* **clear** *clear-percent*<br>**Example:**<br>`switch(config-applet) # event sysmgr memory minor 80` | Triggers an event if the specified system manager memory threshold is exceeded.<br>The *percent* range is from 1 to 99. |
| **Step 9** | **event temperature** [**module** *slot*] [**sensor** *number*] **threshold** {**any** \| **down** \| **up**}<br>**Example:**<br>`switch(config-applet) # event temperature module 2 threshold any` | Triggers an event if the temperature sensor exceeds the configured threshold.<br>The *sensor* range is from 1 to 18. |
| **Step 10** | **event track** [**tag** *tag*] *object-number* **state** {**any** \| **down** \| **up**<br>**Example:**<br>`switch(config-applet) # event track 1 state down` | Triggers an event if the tracked object is in the configured state.<br>The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy.<br>The *object-number* range is from 1 to 500. |

### What to do next

Configure action statements.

If you have already configured action statements or choose not to, complete any of the optional tasks:

- Define a policy using a VSH script. Then, register and activate a VSH script policy.
- Configure memory thresholds
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

# Configuring Action Statements

You can configure an action by using one of the following commands in EEM configuration mode (config-applet):

**Note** If you want 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 command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute. You can use the **terminal event-manager bypass** command to allow all EEM policies with matches to execute the command.

### Before you begin

Define a user policy.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **action** *number*[.*number2*] **cli** *command1*[*command2*.] [**local**]<br>**Example:**<br>`switch(config-applet) # action 1.0 cli "show interface e 3/1"` | Runs the configured commands. You can optionally run the commands on the module where the event occurred.<br>The action label is in the format number1.number2.<br>The *number* can be any number from 1 to 16 digits.<br>The range for *number2* is from 0 to 9. |
| **Step 2** | **action** *number*[.*number2*] **counter name** *counter* **value** *val* **op** {**dec** \| **inc** \| **nop** \| **set**}<br>**Example:**<br>`switch(config-applet) # action 2.0 counter name mycounter value 20 op inc` | Modifies the counter by the configured value and operation.<br>The action label is in the format number1.number2.<br>The *number* can be any number from 1 to 16 digits.<br>The range for *number2* is from 0 to 9.<br>The *counter* can be any case-sensitive, alphanumeric string up to 28 characters. |

| | Command or Action | Purpose |
|---|---|---|
| | | The *val* can be an integer from 0 to 2147483647 or a substituted parameter. |
| **Step 3** | **action** *number*[.*number2*] **event-default**<br>**Example:**<br>`switch(config-applet) # action 1.0 event-default` | Completes the default action for the associated event.<br>The action label is in the format number1.number2.<br>The *number* can be any number from 1 to 16 digits.<br>The range for *number2* is from 0 to 9. |
| **Step 4** | **action** *number*[.*number2*] **policy-default**<br>**Example:**<br>`switch(config-applet) # action 1.0 policy-default` | Completes the default action for the policy that you are overriding.<br>The action label is in the format number1.number2.<br>The *number* can be any number from 1 to 16 digits.<br>The range for *number2* is from 0 to 9. |
| **Step 5** | **action** *number*[.*number2*] **reload** [**module** *slot* [- *slot*]]<br>**Example:**<br>`switch(config-applet) # action 1.0 reload module 3-5` | Forces one or more modules to the entire system to reload.<br>The action label is in the format number1.number2.<br>The *number* can be any number from 1 to 16 digits.<br>The range for *number2* is from 0 to 9. |
| **Step 6** | **action** *number*[.*number2*] **snmp-trap** [**intdata1** *integer-data1*] [**intdata2** *integer-data2*] [**strdata** *string-data*]<br>**Example:**<br>`switch(config-applet) # action 1.0 snmp-trap strdata "temperature problem"` | Sends an SNMP trap with the configured data. The action label is in the format number1.number2.<br>The *number* can be any number from 1 to 16 digits.<br>The range for *number2* is from 0 to 9.<br>The *data* elements can be any number up to 80 digits.<br>The *string* can be any alphanumeric string up to 80 characters. |
| **Step 7** | **action** *number*[.*number2*] **syslog** [**priority** *prio-val*] **msg** *error-message*<br>**Example:**<br>`switch(config-applet) # action 1.0 syslog priority notifications msg "cpu high"` | Sends a customized syslog message at the configured priority.<br>The action label is in the format number1.number2. |

| | Command or Action | Purpose |
|---|---|---|
| | | The *number* can be any number from 1 to 16 digits.<br>The range for *number2* is from 0 to 9.<br>The *error-message* can be any quoted alphanumeric string up to 80 characters. |

### What to do next

Configure event statements.

If you have already configured event statements or choose not to, complete any of the optional tasks:

- Define a policy using a VSH script. Then, register and activate a VSH script policy.
- Configure memory thresholds
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

# Defining a Policy Using a VSH Script

This is an optional task. Complete the following steps if you are using a VSH script to write EEM policies:

### 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

### What to do next

Register and activate a VSH script policy.

# Registering and Activating a VSH Script Policy

This is an optional task. Complete the following steps if you are using a VSH script to write EEM policies.

### Before you begin

Define a policy using a VSH script and copy the file to the system directory.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **event manager policy** *policy-script*<br>**Example:**<br>`switch(config)# event manager policy moduleScript` | Registers and activates an EEM script policy.<br>The *policy-script* can be any case-sensitive, alphanumeric string up to 29 characters. |
| **Step 3** | (Optional) **event manager policy internal** *name*<br>**Example:**<br>`switch(config)# event manager policy internal moduleScript` | Registers and activates an EEM script policy.<br>The *policy-script* can be any case-sensitive alphanumeric string up to 29 characters. |
| **Step 4** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### What to do next

Complete any of the following, depending on your system requirements:

- Configure memory thresholds.
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

# Overriding a System Policy

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | (Optional) **show event manager policy-state** *system-policy* | Displays information about the system policy that you want to override, including thresholds. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`switch(config-applet)# show event manager policy-state __ethpm_link_flap`<br>`Policy __ethpm_link_flap`<br>`Cfg count : 5`<br>`Cfg time interval : 10.000000 (seconds)`<br>`Hash default, Count 0` | Use the **show event manager system-policy** command to find the system policy names. |
| **Step 3** | **event manager applet** *applet-name* **override** *system-policy*<br>**Example:**<br>`switch(config-applet)# event manager applet`<br>`ethport override __ethpm_link_flap`<br>`switch(config-applet)#` | Overrides a system policy and enters applet configuration mode.<br>The *applet-name* can be any case-sensitive, alphanumeric string up to 80 characters.<br>The *system-policy* must be one of the system policies. |
| **Step 4** | **description** *policy-description*<br>**Example:**<br>`switch(config-applet)# description "Overrides link flap policy"` | Configures a descriptive string for the policy.<br>The *policy-description* can be any case-sensitive, alphanumeric string up to 80 characters, but it must be enclosed in quotation marks. |
| **Step 5** | **event** *event-statement*<br>**Example:**<br>`switch(config-applet)# event policy-default count 2 time 1000` | Configures the event statement for the policy. |
| **Step 6** | **section** *number action-statement*<br>**Example:**<br>`switch(config-applet)# action 1.0 syslog priority warnings msg "Link is flapping."` | Configures an action statement for the policy. For multiple action statements, repeat this step. |
| **Step 7** | (Optional) **show event manager policy-state** *name*<br>**Example:**<br>`switch(config-applet)# show event manager policy-state ethport` | Displays information about the configured policy. |
| **Step 8** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

# Configuring Syslog as an EEM Publisher

Configuring syslog as an EEM publisher allows you to monitor syslog messages from the switch.

**Note** The maximum number of searchable strings to monitor syslog messages is 10.

### Before you begin

- Confirm that EEM is available for registration by the syslog.
- Confirm that the syslog daemon is configured and executed.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **event manager applet** *applet-name*<br>**Example:**<br>`switch(config)# event manager applet abc`<br>`switch (config-appliet)#` | Registers an applet with EEM and enters applet configuration mode. |
| **Step 3** | **event syslog** [**tag** *tag*] {**occurs** *number* \| **period** *seconds* \| **pattern** *msg-text* \| **priority** *priority*}<br>**Example:**<br>`switch(config-applet)# event syslog occurs 10` | Registers an applet with EEM and enters applet configuration mode. |
| **Step 4** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### What to do next

Verify your EEM configuration.

# Verifying the Embedded Event Manager Configuration

Use one of the following commands to verify the configuration:

| Command | Purpose |
|---|---|
| **show event manager environment** [*variable-name* \| **all**] | Displays information about the event manager environment variables. |

| Command | Purpose |
|---|---|
| **show event manager event-types** [*event* \| **all** \| **module** *slot*] | Displays information about the event manager event types. |
| **show event manager history events** [**detail**] [**maximum** *num-events*] [**severity** {**catastrophic** \| **minor** \| **moderate** \| **severe**}] | Displays the history of events for all policies. |
| **show event manager policy-state** *policy-name* | Displays information about the policy state, including thresholds. |
| **show event manager script system** [*policy-name* \| **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 Embedded Event Manager

The following example shows how to override the __lcm_module_failure system policy by changing the threshold for only module 3 hitless upgrade failures. It 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
```

The following example shows how to override the __ethpm_link_flap system policy and shut 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
```

The following example shows how to create an EEM policy that allows the 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 does not allow the command to execute.

The following 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
```

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| EEM commands | *Cisco Nexus 3000 Series NX-OS System Management Command Reference* |

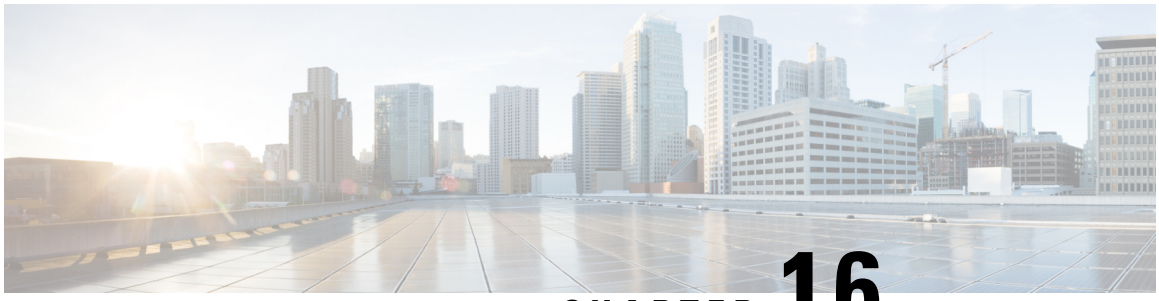
### Standards

There are no new or modified standards supported by this feature, and support for existing standards has not been modified by this feature.

# Feature History for EEM

***Table 28: Feature History for EEM***

| Feature Name | Release | Feature Information |
|---|---|---|
| EEM | 5.0(3)U3(1) | Feature added. |



CHAPTER 16

# Configuring SPAN

This chapter contains the following sections:


- Information About SPAN, on page 181
- SPAN Sources, on page 181
- Characteristics of Source Ports, on page 182
- SPAN Destinations, on page 182
- Characteristics of Destination Ports, on page 182
- Guidelines and Limitations for SPAN, on page 183
- Creating or Deleting a SPAN Session, on page 184
- Configuring an Ethernet Destination Port, on page 185
- Configuring Source Ports, on page 186
- Configuring Source Port Channels or VLANs, on page 187
- Configuring the Description of a SPAN Session, on page 187
- Activating a SPAN Session, on page 188
- Suspending a SPAN Session, on page 188
- Displaying SPAN Information, on page 189
- Configuration Examples for SPAN, on page 190


## Information About SPAN

The Switched Port Analyzer (SPAN) feature (sometimes called port mirroring or port monitoring) selects network traffic for analysis by a network analyzer. The network analyzer can be a Cisco SwitchProbe or other Remote Monitoring (RMON) probes.

## SPAN Sources

SPAN sources refer to the interfaces from which traffic can be monitored. The Cisco Nexus device supports Ethernet, Fibre Channel, virtual Fibre Channel, port channels, SAN port channels, VSANs and VLANs as SPAN sources. With VLANs or VSANs, all supported interfaces in the specified VLAN or VSAN are included as SPAN sources. You can choose the SPAN traffic in the ingress direction, the egress direction, or both directions for Ethernet, Fibre Channel, and virtual Fibre Channel source interfaces:

- Ingress source (Rx)—Traffic entering the device through this source port is copied to the SPAN destination port.

- Egress source (Tx)—Traffic exiting the device through this source port is copied to the SPAN destination port.

You can also configure SPAN source sessions to filter ingress traffic (Rx) by using VLAN access control lists (VACLs).

The Cisco Nexus 34180YC platform switch does not support VLANs as a span source.

# Characteristics of Source Ports

A source port, also called a monitored port, is a switched interface that you monitor for network traffic analysis. The switch supports any number of ingress source ports (up to the maximum number of available ports on the switch) and any number of source VLANs.

A source port has these characteristics:

- Can be of Ethernet, port channel, or VLAN port type.
- Without an ACL filter configured, the same source can be configured for multiple sessions as long as either the direction or SPAN destination is different. However, each SPAN RX source should be configured for only one SPAN session with an ACL filter.
- Cannot be a destination port.
- Can be configured with a direction (ingress, egress, or both) to monitor. For VLAN sources, the monitored direction can only be ingress and applies to all physical ports in the group. The RX/TX option is not available for VLAN SPAN sessions.
- Ingress traffic can be filtered by using ACLs so that they mirror only those packets of information that match the ACL criteria.
- Can be in the same or different VLANs.

# SPAN Destinations

SPAN destinations refer to the interfaces that monitors source ports. The Cisco Nexus Series device supports Ethernet interfaces as SPAN destinations.

# Characteristics of Destination Ports

Each local SPAN session must have a destination port (also called a monitoring port) that receives a copy of traffic from the source ports or VLANs. A destination port has these characteristics:

- Can be any physical port. Source Ethernet and FCoE ports cannot be destination ports.
- Cannot be a source port.
- Cannot be a port channel.
- Does not participate in spanning tree while the SPAN session is active.
- Is excluded from the source list and is not monitored if it belongs to a source VLAN of any SPAN session.

- Receives copies of sent and received traffic for all monitored source ports.

# Guidelines and Limitations for SPAN

SPAN has the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 7.0(3)I4(1), the same source (ethernet or port-channel) can be a part of multiple sessions. You can configure two monitor session with different destinations, but the same source VLAN is not supported.
- The combination of VLAN source session and port source session is not supported. If the traffic stream matches the VLAN source session as well as 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 to the following Cisco devices:

  ***Table 29: Cisco Nexus 3000 Series Switches***

| | | |
|---|---|---|
| Cisco Nexus 3048TP | Cisco Nexus 31128PQ | Cisco Nexus 3132Q |
| Cisco Nexus 3172PQ | Cisco Nexus 3172TQ | Cisco Nexus 3172TQ-XL |

- Beginning with Cisco NX-OS Release 7.0(3)I4(1), multiple ACL filters are supported on the same source.
- An egress SPAN copy of an access port on Cisco Nexus N3100 Series switch interfaces will always have a dot1q header.
- In earlier releases, only tx info was displayed under the **show monitor session** command output. Starting with Release 7.0(3)I2(1), the output of the **show monitor session** command displays all directions for the source VLAN and it does not display any option for the filter VLAN.
- If you install Release NX-OS 5.0(3)U2(2) and then downgrade to a lower version of software, the SPAN configuration is lost.

  You must save the configuration before upgrading to Release NX-OS 5.0(3)U2(2), and then reapply the local span configurations after the downgrade.

  For information about a similar ERSPAN limitation, see Guidelines and Limitations for ERSPAN, on page 195
- ACL filtering is supported only for Rx SPAN. Tx SPAN mirrors all traffics that egresses at the source interface.
- ACL filtering is not supported for IPv6 and MAC ACLs because of ternary content addressable memory (TCAM) width limitations.
- UDF-SPAN acl-filtering only supports source interface rx. This limitation applies to the following switches:
  - Cisco Nexus 3048TP
  - Cisco Nexus 31108TC-V
  - Cisco Nexus 3132Q-40GX

- Cisco Nexus 3132Q-V
- Cisco Nexus 31108PC-V
- Cisco Nexus 3172PQ
- Cisco Nexus 3172TQ
- Cisco Nexus 3164Q
- Cisco Nexus 31128PQ-10GE
- Cisco Nexus 3232C
- Cisco Nexus 3264Q

- The SPAN TCAM size is 128 or 256, depending on the ASIC. One entry is installed as the default and four are reserved for ERSPAN.
- If the same source is configured in more than one SPAN session, and each session has an ACL filter configured, the source interface is programmed only for the first active SPAN session. Hardware entries programmed for ACEs in other sessions is not included in this source interface.
- Both permit and deny access control entries (ACEs) are treated alike. Packets that match the ACE are mirrored irrespective of whether they have a permit or deny entry in the ACL.

**Note** A deny ACE does not result in a dropped packet. An ACL configured in a SPAN session determines only whether the packet is mirrored or not.

- It is recommended to use only the RX type of source traffic for SPAN to provide better performance because RX traffic is cut-through, whereas TX is store-and-forward. Hence, when monitoring both directions (RX and TX), the performance is not as good as when monitoring only RX. If you need to monitor both directions of traffic, you can monitor RX on more physical ports to capture both sides of the traffic.

# Creating or Deleting a SPAN Session

You create a SPAN session by assigning a session number using the **monitor session** command. If the session already exists, any additional configuration information is added to the existing session.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **monitor session** *session-number* | Enters the monitor configuration mode. New session configuration is added to the existing session configuration. |

### Example

The following example shows how to configure a SPAN monitor session:

```
switch# configure terminal
switch(config) # monitor session 2
switch(config) #
```

# Configuring an Ethernet Destination Port

You can configure an Ethernet interface as a SPAN destination port.

**Note** The SPAN destination port can only be a physical port on the switch.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **interface ethernet** *slot*/*port* | Enters interface configuration mode for the Ethernet interface with the specified slot and port.<br>**Note**<br>To enable the **switchport monitor** command on virtual ethernet ports, you can use the **interface vethernet** *slot*/*port* command. |
| **Step 3** | switch(config-if)# **switchport monitor** | Enters monitor mode for the specified Ethernet interface. Priority flow control is disabled when the port is configured as a SPAN destination. |
| **Step 4** | switch(config-if)# **exit** | Reverts to global configuration mode. |
| **Step 5** | switch(config)# **monitor session** *session-number* | Enters monitor configuration mode for the specified SPAN session. |
| **Step 6** | switch(config-monitor)# **destination interface ethernet** *slot*/*port* | Configures the Ethernet SPAN destination port.<br>**Note**<br>To enable the virtual ethernet port as destination interface in the monitor configuration, you can use the **destination interface vethernet** *slot*/*port* command. |

### Example

The following example shows how to configure an Ethernet SPAN destination port (HIF):

```
switch# configure terminal
switch(config)# interface ethernet100/1/24
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 1
switch(config-monitor)# destination interface ethernet100/1/24
switch(config-monitor)#
```

The following example shows how to configure a virtual ethernet (VETH) SPAN destination port:

```
switch# configure terminal
switch(config)# interface vethernet10
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 2
switch(config-monitor)# destination interface vethernet10
switch(config-monitor)#
```

# Configuring Source Ports

Source ports can only be Ethernet ports.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **monitor session** *session-number* | Enters monitor configuration mode for the specified monitoring session. |
| **Step 3** | switch(config-monitor) # **source interface** *type slot/port* [**rx** \| **tx** \| **both**] | Adds an Ethernet SPAN source port and specifies the traffic direction in which to duplicate packets. You can enter a range of Ethernet, Fibre Channel, or virtual Fibre Channel ports. You can specify the traffic direction to duplicate as ingress (Rx), egress (Tx), or both. By default, the direction is both. |

### Example

The following example shows how to configure an Ethernet SPAN source port:

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# filter access-group acl1
switch(config-monitor)# source interface ethernet 1/16
switch(config-monitor)#
```

# Configuring Source Port Channels or VLANs

You can configure the source channels for a SPAN session. These ports can be port channels and VLANs. The monitored direction can be ingress, egress, or both and applies to all physical ports in the group.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **monitor session** *session-number* | Enters monitor configuration mode for the specified SPAN session. |
| **Step 3** | switch(config-monitor) # **filter access-group** *access-map* | Filters ingress traffic at source ports based on the ACL list. Only packets that match the access-list used by access-map are spanned. |
| **Step 4** | switch(config-monitor) # **source** {**interface** {**port-channel**} *channel-number* [**rx** \| **tx** \| **both**] \| **vlan** *vlan-range*} | Configures port channel or VLAN sources. For VLAN sources, the monitored direction is implicit. |

### Example

The following example shows how to configure a port channel SPAN source:

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# filter access-group acl1
switch(config-monitor)# source interface port-channel 1 rx
switch(config-monitor)# source interface port-channel 3 tx
switch(config-monitor)# source interface port-channel 5 both
switch(config-monitor)#
```

The following example shows how to configure a VLAN SPAN source:

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# filter access-group acl1
switch(config-monitor)# source vlan 1
switch(config-monitor)#
```

# Configuring the Description of a SPAN Session

For ease of reference, you can provide a descriptive name for a SPAN session.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **monitor session** *session-number* | Enters monitor configuration mode for the specified SPAN session. |
| **Step 3** | switch(config-monitor) # **description** *description* | Creates a descriptive name for the SPAN session. |

#### Example

The following example shows how to configure a SPAN session description:

```
switch# configure terminal
switch(config) # monitor session 2
switch(config-monitor) # description monitoring ports eth2/2-eth2/4
switch(config-monitor) #
```

# Activating a SPAN Session

The default is to keep the session state shut. You can open a session that duplicates packets from sources to destinations.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **no monitor session** {**all** \| *session-number*} **shut** | Opens the specified SPAN session or all sessions. |

#### Example

The following example shows how to activate a SPAN session:

```
switch# configure terminal
switch(config) # no monitor session 3 shut
```

# Suspending a SPAN Session

By default, the session state is **shut**.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **monitor session** {**all** \| *session-number*} **shut** | Suspends the specified SPAN session or all sessions. |

#### Example

The following example shows how to suspend a SPAN session:

```
switch# configure terminal
switch(config) # monitor session 3 shut
switch(config) #
```

# Displaying SPAN Information

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **show monitor** [**session** {**all** \| *session-number* \| **range** *session-range*} [**brief**]] | Displays the SPAN configuration. |

#### Example

The following example shows how to display SPAN session information:

```
switch# show monitor
SESSION  STATE        REASON                  DESCRIPTION
-------  -----------  ----------------------  --------------------------------
2        up           The session is up
3        down         Session suspended
4        down         No hardware resource
```

The following example shows how to display SPAN session details:

```
switch# show monitor session 2
   session 2
---------------
type              : local
state             : up

source intf       :

source VLANs      :
    rx            : 100
    tx            :
    both          :
filter VLANs      : filter not specified
destination ports : Eth3/1
```

# 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
```

## 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

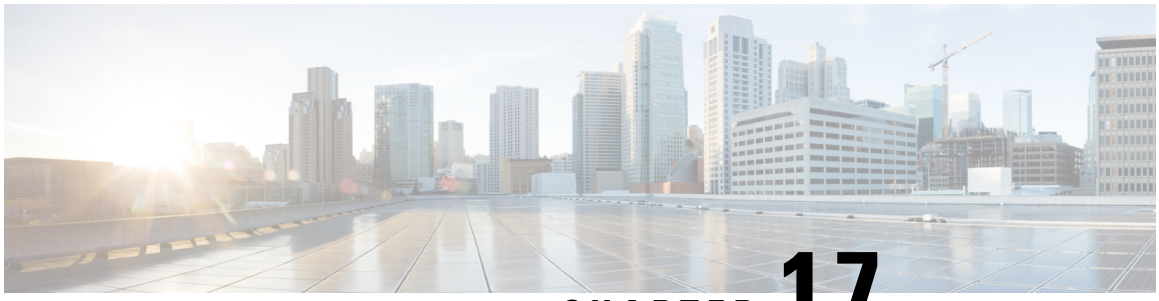
- UDF match value: 0x20
- UDF mask: 0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify 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
  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 l4 26 2
udf udf_pktsig_lsb header outer l4 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
  source interface Ethernet 1/1
  filter access-group acl-udf-pktsig
```



CHAPTER 17

# Configuring Local SPAN and ERSPAN

This chapter contains the following sections:


- Information About ERSPAN, on page 193
- Prerequisites for ERSPAN, on page 195
- Guidelines and Limitations for ERSPAN, on page 195
- Default Settings for ERSPAN, on page 199
- Configuring ERSPAN, on page 199
- Configuration Examples for ERSPAN, on page 211
- Additional References, on page 213


# Information About ERSPAN

The Cisco NX-OS system supports the Encapsulated Remote Switching Port Analyzer (ERSPAN) feature on both source and destination ports. ERSPAN transports mirrored traffic over an IP 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.

ERSPAN consists of an ERSPAN source session, routable ERSPAN generic routing encapsulation (GRE)-encapsulated traffic, and an ERSPAN destination session. You can separately configure ERSPAN source sessions and destination sessions on different switches. You can also configure ERSPAN source sessions to filter ingress traffic by using ACLs.

## 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 and port channels.
- VLANs—When a VLAN is specified as an ERSPAN source, all supported interfaces in the VLAN are ERSPAN sources.

ERSPAN source ports have the following characteristics:

- A port configured as a source port cannot also be configured as a destination port.
- ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.

- Ingress traffic at source ports can be filtered by using ACLs so that they mirror only those packets of information that match the ACL criteria.

# ERSPAN Destinations

ERSPAN destination sessions capture packets sent by ERSPAN source sessions on Ethernet ports or port channels and send them to the destination port. Destination ports receive the copied traffic from ERSPAN sources.

ERSPAN destination sessions are identified by the configured source IP address and ERSPAN ID. This allows multiple source sessions to send ERSPAN traffic to the same destination IP and ERSPAN ID and allows you to have multiple sources terminating at a single destination simultaneously.

ERSPAN destination ports have the following characteristics:

- A port configured as a destination port cannot also be configured as a source port.
- Destination ports do not participate in any spanning tree instance or any Layer 3 protocols.
- Ingress and ingress learning options are not supported on monitor destination ports.
- Host Interface (HIF) port channels and fabric port channel ports are not supported as SPAN destination ports.

# ERSPAN Sessions

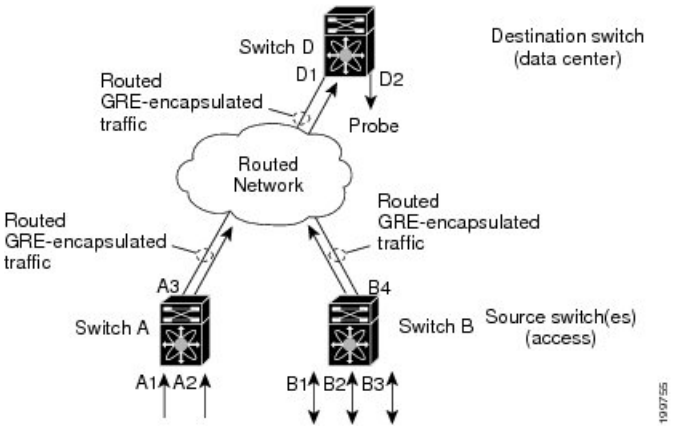
You can create ERSPAN sessions that designate sources and destinations to monitor.

When configuring ERSPAN source sessions, you must configure the destination IP address. When configuring ERSPAN destination sessions, you must configure the source IP address. See ERSPAN Sources, on page 193 for the properties of source sessions and ERSPAN Destinations, on page 194 for the properties of destination sessions.

**Note** Only two ERSPAN or SPAN source sessions can run simultaneously across all switches. Only 23 ERSPAN destination sessions can run simultaneously across all switches.

The following figure shows an ERSPAN configuration.

*Figure 1: ERSPAN Configuration*




## Multiple ERSPAN Sessions

Although you can define up to 18 ERSPAN sessions, only a maximum of four ERSPAN or SPAN sessions can be operational simultaneously. If both receive and transmit sources are configured in the same session, only two ERSPAN or SPAN sessions can be operational simultaneously. You can shut down any unused ERSPAN sessions.

**Note** The Cisco Nexus 34180YC platform switch supports a total of 32 sessions SPAN and ERSPAN sessions together configured on the switch and, all 32 can be active at the same time.

For information about shutting down ERSPAN sessions, see Shutting Down or Activating an ERSPAN Session, on page 209.

## High Availability

The ERSPAN feature supports stateless and stateful restarts. After a reboot or supervisor switchover, the running configuration is applied.

# Prerequisites for ERSPAN

ERSPAN has the following prerequisite:

•You must first configure the Ethernet interfaces for ports on each device to support the desired ERSPAN configuration. For more information, see the Interfaces configuration guide for your platform.

# Guidelines and Limitations for ERSPAN

ERSPAN has the following configuration guidelines and limitations:

- Beginning with Cisco NX-OS Release 7.0(3)I4(1), the same source can be part of multiple sessions.

- Beginning with Cisco NX-OS Release 7.0(3)I4(1), multiple ACL filters are supported on the same source.
- Two ERSPAN destination sessions are not supported on Cisco Nexus 3000, 3100, and 3200 platform switches.
- ERSPAN supports the following:
  - From 4 to 6 tunnels
  - Nontunnel packets
  - IP-in-IP tunnels
  - IPv4 tunnels (limited)
  - Cisco Nexus 3000 Series switches use a generic GRE ERSPAN header format for spanning packets matching ERSPAN source session. This format does not conform to the Cisco ERSPAN Type 1/2/3 header format. Cisco ASIC based platforms support ERSPAN termination and decapsulation only for ERSPAN packets conforming to Cisco ERSPAN encapsulation format Type. Hence, ERSPAN packets originating from Cisco Nexus 3000 Series switches to the local destination IP address of the CISCO ASIC based switch will not match the ERSPAN termination filter; If the destination IP address is also the local IP address on the Cisco ASIC platform, the ERSPAN packets are sent to software and dropped in software.
  - ERSPAN destination session type (however, support for decapsulating the ERSPAN packet is not available. The entire encapsulated packet is spanned to a front panel port at the ERSPAN terminating point.)
- ERSPAN packets are dropped if the encapsulated mirror packet fails Layer 2 MTU checks.
- There is a 112-byte limit for egress encapsulation. Packets that exceed this limit are dropped. This scenario might be encountered when tunnels and mirroring are intermixed.
- ERSPAN sessions are shared with local sessions. A maximum of 18 sessions can be configured; however only a maximum of four sessions can be operational at the same time. If both receive and transmit sources are configured in the same session, only two sessions can be operational.
- If you install Release NX-OS 5.0(3)U2(2), configure ERSPAN, and then downgrade to a lower version of software, the ERSPAN configuration is lost. This situation occurs because ERSPAN is not supported in versions before Release NX-OS 5.0(3)U2(2).

  For information about a similar SPAN limitation, see Guidelines and Limitations for SPAN, on page 183.
- ERSPAN and ERSPAN ACLs are not supported for packets that are generated by the supervisor.
- ERSPAN and ERSPAN with ACL filtering are not supported for packets generated by the supervisor.
- ACL filtering is supported only for Rx ERSPAN. Tx ERSPAN that mirrors all traffic egressed at the source interface.
- ACL filtering is not supported for IPv6 and MAC ACLs because of TCAM width limitations.
- If the same source is configured in more than one ERSPAN session, and each session has an ACL filter configured, the source interface will be programmed only for the first active ERSPAN session. The ACEs that belong to the other sessions will not have this source interface programmed.

- If you configure an ERSPAN session and a local SPAN session (with filter access-group and allow-sharing option) to use the same source, the local SPAN session goes down when you save the configuration and reload the switch.
- The drop action is not supported with the VLAN access-map configuration with the filter access-group for a monitor session. The monitor session goes into an error state if the VLAN access-map with a drop action is configured with the filter access-group in the monitor session.
- Both permit and deny ACEs are treated alike. Packets that match the ACE are mirrored irrespective of whether they have a permit or deny entry in the ACL.
- ERSPAN is not supported for management ports.
- A destination port can be configured in only one ERSPAN session at a time.
- You cannot configure a port as both a source and destination port.
- A single ERSPAN session can include mixed sources in any combination of the following:
  - Ethernet ports or port channels but not subinterfaces.
  - VLANs or port channels, which can be assigned to port channel subinterfaces.
  - Port channels to the control plane CPU.

**Note** ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.

- Destination ports do not participate in any spanning tree instance or Layer 3 protocols.
- When an ERSPAN session contains source ports that are monitored in the transmit or transmit and receive direction, packets that these ports receive may be replicated to the ERSPAN destination port even though 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
- For VLAN ERSPAN sessions with both ingress and egress configured, two packets (one from ingress and one from egress) are forwarded from the destination port if the packets get switched on the same VLAN.
- VLAN ERSPAN monitors only the traffic that leaves or enters Layer 2 ports in the VLAN.
- When the Cisco Nexus 3000 series switch is the ERSPAN destination, GRE headers are not stripped off before sending mirrored packets out of the terminating point. Packets are sent along with the GRE headers as GRE packets and the original packet as the GRE payload.
- You can view the SPAN/ERSPAN ACL statistics using the **show monitor filter-list** command. The output of the command displays all the entries along with the statistics from the SPAN TCAM. The ACL name is not printed, but only the entries are printed in the output. You can clear the statistics using the **clear monitor filter-list statistics** command. The output is similar to **show ip access-list** command. The Cisco Nexus 3000 series switch does not provide support per ACL level statistics. This enhancement is supported for both local SPAN and ERSPAN.

- The traffic to and/or from the CPU is spanned. It is similar to any other interface SPAN. This enhancement is supported only in local SPAN. It is not supported with ACL source. The Cisco Nexus 3000 series switch does not span the packets with (RCPU.dest_port != 0) header that is sent out from the CPU.
- For SPAN forward drop traffic, SPAN only the packets that get dropped due to various reasons in the forwarding plane. This enhancement is supported only for ERSPAN Source session. It is not supported along with SPAN ACL, Source VLAN, and Source interface. Three ACL entries are installed to SPAN dropped traffic. Priority can be set for the drop entries to have a higher/lower priority than the SPAN ACL entries and the VLAN SPAN entries of the other monitor sessions. By default, the drop entries have a higher priority.
- SPAN UDF (User Defined Field) based ACL support
  - You can match any packet header or payload (certain length limitations) in the first 128 bytes of the packet.
  - You can define the UDFs with particular offset and length to match.
  - You can match the length as 1 or 2 bytes only.
  - Maximum of 8 UDFs are supported.
  - Additional UDF match criteria is added to ACL.
  - The UDF match criteria can be configured only for SPAN ACL. This enhancement is not supported for other ACL features, for example, RACL, PACL, and VACL.
  - Each ACE can have up to 8 UDF match criteria.
  - The UDF and http-redirect configuration should not co-exist in the same ACL.
  - The UDF names need to be qualified for the SPAN TCAM.
  - The UDFs are effective only if they are qualified by the SPAN TCAM.
  - The configuration for the UDF definition and the UDF name qualification in the SPAN TCAM require the use of **copy r s** command and reload.
  - The UDF match is supported for both Local SPAN and ERSPAN Src sessions.
  - The UDF name can have a maximum length of 16 characters.
  - The UDF offset starts from 0 (zero). If offset is specified as an odd number, 2 UDFs are used in the hardware for one UDF definition in the software. The configuration is rejected if the number of UDFs usage in the hardware goes beyond 8.
  - The UDF match requires the SPAN TCAM region to go double-wide. Therefore, you have to reduce the other TCAM regions' size to make space for SPAN.
  - The SPAN UDFs are not supported in tap-aggregation mode.
- If a sup-eth source interface is configured in the erspan-src session, the acl-span cannot be added as a source into that session and vice-versa.
- ERSPAN source and ERSPAN destination sessions must use dedicated loopback interfaces. Such loopback interfaces should not be having any control plane protocols.

# Default Settings for ERSPAN

The following table lists the default settings for ERSPAN parameters.

*Table 30: Default ERSPAN Parameters*

| Parameters | Default |
|---|---|
| ERSPAN sessions | Created in the shut state. |

# Configuring ERSPAN

## 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.

For sources, you can specify Ethernet ports, port channels, and VLANs. A single ERSPAN session can include mixed sources in any combination of Ethernet ports or VLANs.

**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**<br>**Example:**<br>`switch# config t`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **monitor erspan origin ip-address** *ip-address* **global**<br>**Example:**<br>`switch(config)# monitor erspan origin ip-address 10.0.0.1 global` | Configures the ERSPAN global origin IP address. |
| **Step 3** | **no monitor session** {*session-number* \| **all**}<br>**Example:**<br>`switch(config)# no monitor session 3` | 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** | Configures an ERSPAN source session. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`switch(config)# monitor session 3 type erspan-source`<br>`switch(config-erspan-src)#` | |
| **Step 5** | **description** *description*<br>**Example:**<br>`switch(config-erspan-src)# description erspan_src_session_3` | Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters. |
| **Step 6** | **filter access-group** *acl-name*<br>**Example:**<br>`switch(config-erspan-src)# filter access-group acl1` | Filters ingress traffic at source ports based on the ACL list. Only packets that match the access list are spanned. The *acl-name* is an IP access-list, but not an access-map. |
| **Step 7** | **source** {**interface** *type* [**rx** [**allow-pfc**] \| **tx** \| **both**] \| **vlan** {*number* \| *range*} [**rx**] \| **forward-drops rx** [**priority-low**]}<br>**Example:**<br>`switch(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx`<br>**Example:**<br>`switch(config-erspan-src)# source interface port-channel 2`<br>**Example:**<br>`switch(config-erspan-src)# source interface sup-eth 0 both`<br>**Example:**<br>`switch(config-monitor)# source interface ethernet 101/1/1-3` | Configures the sources and traffic direction in which to copy packets. You can enter a range of Ethernet ports, a port channel, or a range of VLANs.<br>You can configure one or more sources, as either a series of comma-separated entries or a range of numbers. You can specify up to 128 interfaces. For information on the VLAN range, see the *Cisco Nexus 3000 Series NX-OS Layer 2 Switching Configuration Guide*.<br>You can specify the traffic direction to copy as ingress, egress, or both. The default direction is both.<br>The **allow-pfc** option initiates a span of the priority flow control (PFC) frames that are received on a port. PFC frames are allowed in the ingress pipeline instead of being dropped. If ERSPAN is configured for that port, those PFC frames are spanned to the appropriate egress interface. Ports configured with this option can also span normal data traffic.<br>As an alternative to configuring interfaces or VLANs as an ERSPAN source, you can configure ERSPAN to span the maximum number of forward packet drops possible in the ingress pipeline. Doing so can help you to analyze and isolate packet drops in the network. By default, the **source forward-drops rx** command captures packet drops for all ports on the network forwarding module. The **priority-low** option causes this ERSPAN access control entry (ACE) matching drop condition to take a lesser priority to any |

| | Command or Action | Purpose |
|---|---|---|
| | | other ERSPAN ACEs configured by regular interface or VLAN ERSPAN ACLs. |
| **Step 8** | (Optional) Repeat Step 6 to configure all ERSPAN sources. | — |
| **Step 9** | **destination ip** *ip-address*<br>**Example:**<br>`switch(config-erspan-src)# destination ip 10.1.1.1` | Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session. |
| **Step 10** | (Optional) **ip ttl** *ttl-number*<br>**Example:**<br>`switch(config-erspan-src)# ip ttl 25` | Configures the IP time-to-live (TTL) value for the ERSPAN traffic. The range is from 1 to 255. |
| **Step 11** | (Optional) **ip dscp** *dscp-number*<br>**Example:**<br>`switch(config-erspan-src)# ip dscp 42` | Configures the differentiated services code point (DSCP) value of the packets in the ERSPAN traffic. The range is from 0 to 63. |
| **Step 12** | **no shut**<br>**Example:**<br>`switch(config-erspan-src)# no shut` | Enables the ERSPAN source session. By default, the session is created in the shut state.<br>**Note**<br>Only two ERSPAN source sessions can be running simultaneously. |
| **Step 13** | (Optional) **show monitor session** {**all** \| *session-number* \| **range** *session-range*}<br>**Example:**<br>`switch(config-erspan-src)# show monitor session 3` | Displays the ERSPAN session configuration. |
| **Step 14** | (Optional) **show running-config monitor**<br>**Example:**<br>`switch(config-erspan-src)# show running-config monitor` | Displays the running ERSPAN configuration. |
| **Step 15** | (Optional) **show startup-config monitor**<br>**Example:**<br>`switch(config-erspan-src)# show startup-config monitor` | Displays the ERSPAN startup configuration. |
| **Step 16** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config-erspan-src)# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

# Configuring SPAN Forward Drop Traffic for ERSPAN Source Session

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# config t`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **monitor session** {*session-number* \| **all**} **type erspan-source**<br>**Example:**<br>`switch(config)# monitor session 1 type erspan-source`<br>`switch(config-erspan-src)#` | Configures an ERSPAN source session. |
| **Step 3** | **vrf** *vrf-name*<br>**Example:**<br>`switch(config-erspan-src)# vrf default` | Configures the VRF that the ERSPAN source session uses for traffic forwarding. |
| **Step 4** | **destination ip** *ip-address*<br>**Example:**<br>`switch(config-erspan-src)# destination ip 10.1.1.1` | Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session. |
| **Step 5** | **source forward-drops rx** [*priority-low*]<br>**Example:**<br>`switch(config-erspan-src)# source forward-drops rx [priority-low]` | Configures the SPAN forward drop traffic for the ERSPAN source session. When configured as a low priority, this SPAN ACE matching drop condition takes less priority over any other SPAN ACEs configured by the interface ACL SPAN or VLAN ACL SPAN. Without the priority-low keyword, these drop ACEs take high priority compared to the regular interface or the VLAN SPAN ACLs. The priority matters only when the packet matching drop ACEs and the interface/VLAN SPAN ACLs are configured. |
| **Step 6** | **no shut**<br>**Example:**<br>`switch(config-erspan-src)# no shut` | Enables the ERSPAN source session. By default, the session is created in the shut state.<br>**Note**<br>Only two ERSPAN source sessions can be running simultaneously. |
| **Step 7** | (Optional) **show monitor session** {**all** \| *session-number* \| **range** *session-range*}<br>**Example:** | Displays the ERSPAN session configuration. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch(config-erspan-src)# show monitor session 3` | |

### Example

```
switch# config t
  switch(config)# monitor session 1 type erspan-source
  switch(config-erspan-src)# vrf default
  switch(config-erspan-src)# destination ip 40.1.1.1
  switch(config-erspan-src)# source forward-drops rx
  switch(config-erspan-src)# no shut
  switch(config-erspan-src)# show monitor session 1

switch# config t
  switch(config)# monitor session 1 type erspan-source
  switch(config-erspan-src)# vrf default
  switch(config-erspan-src)# destination ip 40.1.1.1
  switch(config-erspan-src)# source forward-drops rx  priority-low
  switch(config-erspan-src)# no shut
  switch(config-erspan-src)# show monitor session 1
```

# 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**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **ip access-list** *acl-name*<br>**Example:**<br>`switch(config)# ip access-list erspan-acl`<br>`switch(config-acl)#` | Creates the ERSPAN ACL and enters IP ACL configuration mode. The *acl-name* argument can be up to 64 characters. |
| **Step 3** | [*sequence-number*] {**permit** \| **deny**} *protocol source destination* [**set-erspan-dscp** *dscp-value*] [**set-erspan-gre-proto** *protocol-value*]<br>**Example:** | Creates a rule in the ERSPAN ACL. You can create many rules. The *sequence-number* argument can be a whole number between 1 and 4294967295.<br>The **permit** and **deny** commands support many ways of identifying traffic. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch(config-acl)# permit ip 192.168.2.0/24 any set-erspan-dscp 40 set-erspan-gre-proto 5555` | 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.<br>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.<br>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:<br>• One ERSPAN session with an ACL having a maximum of three ACEs with the **set-erspan-gre-proto** or **set-erspan-dscp** action<br>• 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<br>• 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** | (Optional) **show ip access-lists** *name*<br>**Example:**<br>`switch(config-acl)# show ip access-lists erpsan-acl` | Displays the ERSPAN ACL configuration. |
| **Step 5** | (Optional) **show monitor session** {**all** \| *session-number* \| **range** *session-range*} [**brief**]<br>**Example:**<br>`switch(config-acl)# show monitor session 1` | Displays the ERSPAN session configuration. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config-acl)# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

# Configuring User Defined Field (UDF) Based ACL Support

You can configure User Defined Field (UDF) based ACL support on Cisco Nexus 3000 Series switches. See the following steps to configure ERSPAN based on UDF. See the Guidelines and Limitations for ERSPAN section for more information.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **udf** < *udf -name*> <*packet start*> <*offset*> <*length*><br>**Example:**<br>`(config)# udf udf1 packet-start 10 2`<br>`(config)# udf udf2 packet-start 50 2` | Defines the UDF.<br>**Note**<br>You can define multiple UDFs but it is recommended to configure only the required UDFs. This configuration takes affect only after attaching the UDFs to a TCAM region and rebooting the box, as the UDFs are added to a region's qualifier set at TCAM carving time (boot up time). |
| **Step 3** | switch(config)# **udf** < *udf -name*> header <*Layer3/Layer4*> <*offset*> <*length*><br>**Example:**<br>`(config)# udf udf3 header outer l4 0 1`<br>`(config)# udf udf3 header outer l4 10 2`<br>`(config)# udf udf3 header outer l4 50 1` | Defines the UDF. |
| **Step 4** | switch(config)# **hardware profile tcam region span qualify udf** <*name1*>…… <*name8*><br>**Example:**<br>`(config)# hardware profile tcam region span qualify udf udf1 udf2 udf3 udf4 udf5`<br>`[SUCCESS] Changes to UDF qualifier set will be applicable only after reboot.`<br>`You need to 'copy run start' and 'reload'`<br>`config)#` | Configure UDF Qualification in SPAN TCAM. Add the UDFs to qualifier set for a TCAM region at TCAM carving time (happens at boot up time). The configuration allows maximum 4 UDFs that can be attached to a span region, all UDFs listed in a single command for a region. A new configuration for a region replaces the current configuration, but note that it needs a reboot for the configuration to come to the effect.<br>When the UDF qualifier is added to the SPAN TCAM, the TCAM region expands from single wide to double wide. Make sure enough free |

| | Command or Action | Purpose |
|---|---|---|
| | | space (128 more single wide entries) is available for the expansion or else the command gets rejected. Re-enter the command after creating the space by reducing TCAM space from the unused regions. Once the UDFs are detached from SPAN/TCAM region using the **no hardware profile tcam region span qualify udf <name1> ..<name8>** command, the SPAN TCAM region is considered as a single wide entry. |
| **Step 5** | switch(config)# **permit** …… <*regular ACE match criteria*> **udf** <*name1*> < *val* > <*mask*> ..... <*name8*> < *val* > <*mask*><br>**Example:**<br>`(config)# ip access-list test`<br>`10 permit ip any any udf udf1 0x1234 0xffff udf3 0x56 0xff`<br>`30 permit ip any any dscp af11 udf udf5 0x22 0x22`<br>`config)#` | Configure an ACL with UDF match. |
| **Step 6** | switch(config)# **show monitor session** <*session-number*><br>**Example:**<br>`(config)# show monitor session 1`<br>`session 1`<br>`---------------`<br>`type : erspan-source`<br>`state : up`<br>`vrf-name : default`<br>`destination-ip : 40.1.1.1`<br>`ip-ttl : 255`<br>`ip-dscp : 0`<br>`acl-name : test`<br>`origin-ip : 100.1.1.10 (global)`<br>`source intf :`<br>`rx : Eth1/20`<br>`tx : Eth1/20`<br>`both : Eth1/20`<br>`source VLANs :`<br>`filter VLANs : filter not specified`<br>`rx :`<br>`source fwd drops :`<br>`egress-intf : Eth1/23`<br>`switch#`<br>`config)#` | Displays the ACL using the **show monitor session <session-number>** command. You can check if the SPAN TCAM region is carved or not using the BCM SHELL command. |

# Configuring IPv6 User Defined Field (UDF) on ERSPAN

You can configure IPv6 User Defined Field (UDF) on ERSPAN on Cisco Nexus 3000 Series switches. See the following steps to configure ERSPAN based on IPv6 UDF. See the Guidelines and Limitations for ERSPAN section for more information

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **udf** < *udf -name*> <*packet start*> <*offset*> <*length*><br>**Example:**<br>`(config)# udf udf1 packet-start 10 2`<br>`(config)# udf udf2 packet-start 50 2` | Defines the UDF.<br>**Note**<br>You can define multiple UDFs but it is recommended to configure only the required UDFs. This configuration takes affect only after attaching the UDFs to a TCAM region and rebooting the box, as the UDFs are added to a region's qualifier set at TCAM carving time (boot up time). |
| **Step 3** | switch(config)# **udf** < *udf -name*> header <*Layer3/Layer4*> <*offset*> <*length*><br>**Example:**<br>`(config)# udf udf3 header outer l4 0 1`<br>`(config)# udf udf3 header outer l4 10 2`<br>`(config)# udf udf3 header outer l4 50 1` | Defines the UDF. |
| **Step 4** | switch(config)# **hardware profile tcam region ipv6-span-l2 512**<br>**Example:**<br>`(config)# hardware profile tcam region ipv6-span-l2 512`<br>`Warning: Please save config and reload the system for the configuration to take effect.`<br>`config)#` | Configure IPv6 on UDF on layer 2 ports. A new configuration for a region replaces the current configuration and you must reboot the switch for the configuration to come to the effect. |
| **Step 5** | switch(config)# **hardware profile tcam region ipv6-span 512**<br>**Example:**<br>`(config)# hardware profile tcam region ipv6-span 512`<br>`Warning: Please save config and reload the system for the configuration to take effect.`<br>`config)#` | Configure IPv6 on UDF on layer 3 ports. A new configuration for a region replaces the current configuration and you must reboot the switch for the configuration to come to the effect. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | switch(config)# **hardware profile tcam region span spanv6 qualify udf** *<name1>…… <name8>*<br>**Example:**<br>`(config)# hardware profile tcam region spanv6 qualify udf udf1`<br>`[SUCCESS] Changes to UDF qualifier set will be applicable only after reboot.`<br>`You need to 'copy run start' and 'reload'`<br>`config)#` | Configure UDF Qualification in SPAN for layer 3 ports. This enables the UDF match for ipv6-span TCAM region. Add the UDFs to qualifier set for a TCAM region at TCAM carving time (happens at boot up time). The configuration allows maximum of 2 IPv6 UDFs that can be attached to a SPAN region, all UDFs listed in a single command for a region. A new configuration for a region replaces the current configuration, but note that it needs a reboot for the configuration to come to the effect. |
| **Step 7** | switch(config)# **hardware profile tcam region span spanv6-12 qualify udf** *<name1>…… <name8>*<br>**Example:**<br>`(config)# hardware profile tcam region spanv6-l2 qualify udf udf1`<br>`[SUCCESS] Changes to UDF qualifier set will be applicable only after reboot.`<br>`You need to 'copy run start' and 'reload'`<br>`config)#` | Configure UDF Qualification in SPAN for layer 2 ports. This enables the UDF match for ipv6-span-12 TCAM region. Add the UDFs to qualifier set for a TCAM region at TCAM carving time (happens at boot up time). The configuration allows a maximum of 2 IPv6 UDFs that can be attached to a SPAN region, all UDFs listed in a single command for a region. A new configuration for a region replaces the current configuration, but note that it needs a reboot for the configuration to come to the effect. |
| **Step 8** | switch (config-erspan-src)# **filter** …… *ipv6 access-group…. <aclname>…. <allow-sharing>*<br>**Example:**<br>`(config-erspan-src)# ipv6 filter access-group test`<br>`(config)#` | Configure a IPv6 ACL in SPAN and ERSPAN mode. You can have only one of “filter ip access-group” or “filter ipv6 access-group” configuration in one monitor session. If same source interface is part of a IPv4 and IPv6 ERSPAN ACL monitor session, the “allow-sharing” needs to be configured with the “filter [ipv6] access-group” in the monitor session configuration. |
| **Step 9** | switch(config)# **permit** …… *<regular ACE match criteria>* **udf** *<name1> < val > <mask> ..... <name8> < val > <mask>*<br>**Example:**<br>`(config-erspan-src)# ipv6 access-list test`<br>`(config-ipv6-acl)# permit ipv6 any any udf udf1 0x1 0x0` | Configure an ACL with UDF match. |
| **Step 10** | switch(config)# **show monitor session** *<session-number>*<br>**Example:** | Displays the ACL using the **show monitor session <session-number>** command. |

| | Command or Action | Purpose |
|---|---|---|
| | (config)# **show monitor session 1**<br>session 1<br>---------------<br>type : erspan-source<br>state : up<br>vrf-name : default<br>destination-ip : 40.1.1.1<br>ip-ttl : 255<br>ip-dscp : 0<br>acl-name : test<br>origin-ip : 100.1.1.10 (global)<br>source intf :<br>rx : Eth1/20<br>tx : Eth1/20<br>both : Eth1/20<br>source VLANs :<br>filter VLANs : filter not specified<br><br>rx :<br>source fwd drops :<br>egress-intf : Eth1/23<br>switch#<br>config)# | |

# Shutting Down or Activating an ERSPAN Session

You can shut down ERSPAN sessions to discontinue the copying of packets from sources to destinations. Because only a specific number of ERSPAN sessions can be running simultaneously, you can shut down a session 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** | **configuration terminal**<br>**Example:**<br>`switch# configuration terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **monitor session** {*session-range* \| **all**} **shut**<br>**Example:**<br>`switch(config)# monitor session 3 shut` | Shuts down the specified ERSPAN sessions. The session range is from 1-18. By default, sessions are created in the shut state. Four unidirectional sessions, or two bidirectional sessions can be active at the same time.<br>**Note**<br>• In Cisco Nexus 5000 and 5500 platforms, two sessions can run simultaneously. |

| | Command or Action | Purpose |
|---|---|---|
| | | • In Cisco Nexus 5600 and 6000 platforms, 16 sessions can run simultaneously. |
| **Step 3** | **no monitor session** {*session-range* \| **all**} **shut**<br>**Example:**<br>`switch(config)# no monitor session 3 shut` | Resumes (enables) the specified ERSPAN sessions. The session range is from 1-18. By default, sessions are created in the shut state. Four unidirectional sessions, or two bidirectional sessions can be active at the same time.<br>**Note**<br>If a monitor session is enabled but its operational status is down, then to enable the session, you must first specify the **monitor session shut** command followed by the **no monitor session shut** command. |
| **Step 4** | **monitor session** *session-number* **type erspan-source**<br>**Example:**<br>`switch(config)# monitor session 3 type erspan-source`<br>`switch(config-erspan-src)#` | Enters the monitor configuration mode for the ERSPAN source type. The new session configuration is added to the existing session configuration. |
| **Step 5** | **monitor session** *session-number* **type erspan-destination**<br>**Example:**<br>`switch(config-erspan-src)# monitor session 3 type erspan-destination` | Enters the monitor configuration mode for the ERSPAN destination type. |
| **Step 6** | **shut**<br>**Example:**<br>`switch(config-erspan-src)# shut` | Shuts down the ERSPAN session. By default, the session is created in the shut state. |
| **Step 7** | **no shut**<br>**Example:**<br>`switch(config-erspan-src)# no shut` | Enables the ERSPAN session. By default, the session is created in the shut state. |
| **Step 8** | (Optional) **show monitor session all**<br>**Example:**<br>`switch(config-erspan-src)# show monitor session all` | Displays the status of ERSPAN sessions. |
| **Step 9** | (Optional) **show running-config monitor**<br>**Example:**<br>`switch(config-erspan-src)# show running-config monitor` | Displays the running ERSPAN configuration. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 10** | (Optional) **show startup-config monitor**<br>**Example:**<br>`switch(config-erspan-src)# show startup-config monitor` | Displays the ERSPAN startup configuration. |
| **Step 11** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config-erspan-src)# copy running-config startup-config` | Copies the running configuration to the startup configuration. |

## Verifying the ERSPAN Configuration

Use the following command to verify the ERSPAN configuration information:

| Command | Purpose |
|---|---|
| **show monitor session** {**all** \| *session-number* \| **range** *session-range*} | Displays the ERSPAN session configuration. |
| **show running-config monitor** | Displays the running ERSPAN configuration. |
| **show startup-config monitor** | Displays the ERSPAN startup configuration. |

# Configuration Examples for ERSPAN

## Configuration Example for an ERSPAN Source Session

The following example shows how to configure an ERSPAN source session:

```
switch# config t
switch(config)# interface e14/30
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# monitor erspan origin ip-address 3.3.3.3 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# filter access-group acl1
switch(config-erspan-src)# source interface e14/30
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 9.1.1.2
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_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 erspan_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 erspan_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 type erspan-source
switch(config-erspan-src)# filter access_group erspan_filter
```

# 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

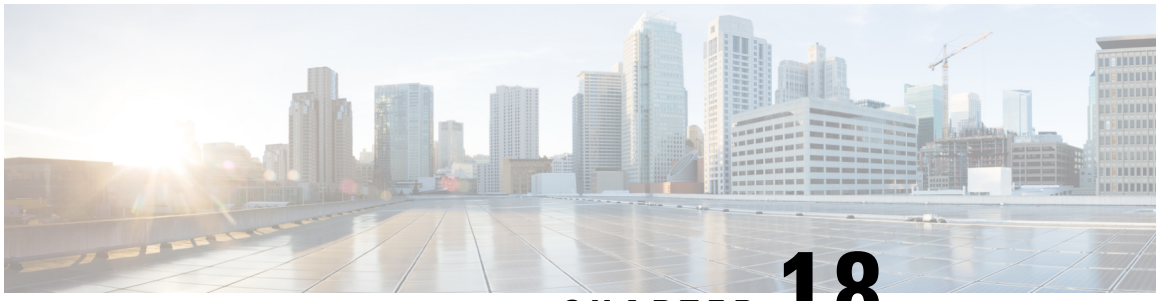
- UDF match value: 0xDEADBEEF (split into two-byte chunks and two UDFs)
- UDF mask: 0xFFFFFFFF

```
udf udf_pktsig_msb header outer l3 26 2
udf udf_pktsig_lsb header outer l3 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
```

# Additional References

## Related Documents

| Related Topic | Document Title |
|---|---|
| ERSPAN commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples | *Cisco Nexus NX-OS System Management Command Reference* for your platform. |



CHAPTER 18

# Configuring DNS

This chapter contains the following sections:


- Information About DNS Client , on page 215
- Prerequisites for DNS Clients, on page 216
- Default Settings for DNS Clients, on page 216
- Configuring the DNS Source Interface, on page 216
- Configuring DNS Clients, on page 217


# Information About DNS Client

If your network devices require connectivity with devices in networks for which you do not control name assignment, you can assign device names that uniquely identify your devices within the entire internetwork using the domain name server (DNS). DNS uses a hierarchical scheme for establishing hostnames for network nodes, which allows local control of the segments of the network through a client-server scheme. The DNS system can locate a network device by translating the hostname of the device into its associated IP address.

On the Internet, a domain is a portion of the naming hierarchy tree that refers to general groupings of networks based on the organization type or geography. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco is a commercial organization that the Internet identifies by a com domain, so its domain name is cisco.com. A specific hostname in this domain, the File Transfer Protocol (FTP) system, for example, is identified as ftp.cisco.com.

## Name Servers

Name servers keep track of domain names and know the parts of the domain tree for which they have complete information. A name server may also store information about other parts of the domain tree. To map domain names to IP addresses in Cisco NX-OS, you must first identify the hostnames, then specify a name server, and enable the DNS service.

Cisco NX-OS allows you to statically map IP addresses to domain names. You can also configure Cisco NX-OS to use one or more domain name servers to find an IP address for a hostname.

## DNS Operation

A name server handles client-issued queries to the DNS server for locally defined hosts within a particular zone as follows:

- An authoritative name server responds to DNS user queries for a domain name that is under its zone of authority by using the permanent and cached entries in its own host table. If the query is for a domain name that is under its zone of authority but for which it does not have any configuration information, the authoritative name server replies that no such information exists.
- A name server that is not configured as the authoritative name server responds to DNS user queries by using information that it has cached from previously received query responses. If no router is configured as the authoritative name server for a zone, queries to the DNS server for locally defined hosts receive nonauthoritative responses.

Name servers answer DNS queries (forward incoming DNS queries or resolve internally generated DNS queries) according to the forwarding and lookup parameters configured for the specific domain.

## High Availability

Cisco NX-OS supports stateless restarts for the DNS client. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

# Prerequisites for DNS Clients

The DNS client has the following prerequisites:

- You must have a DNS name server on your network.

# Default Settings for DNS Clients

The following table shows the default settings for DNS client parameters.

| Parameter | Default |
|---|---|
| DNS client | Enabled |

# Configuring the DNS Source Interface

You can configure DNS to use a specific interface.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **ip dns source-interface** *type slot/port* | Configures the source interface for all DNS packets. The following list contains the valid values for *interface*.<br>• ethernet |

| | Command or Action | Purpose |
|---|---|---|
| | | • loopback<br>• mgmt<br>• port-channel<br>• vlan<br>**Note**<br>When you, configure the source interface for DNS, SCP copy operations intiated from the server fail. To perform an SCP copy operation from the server, remove the DNS source interface configuration. |
| **Step 3** | switch(config)# **show ip dns source-interface** | Displays the configured DNS source interface. |

### Example

This example shows how to configure the DNS source interface:

```
switch(config)# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip dns source-interface ethernet 1/8
switch(config)# show ip dns source-interface
VRF Name                                Interface
default                                 Ethernet1/8
```

# Configuring DNS Clients

You can configure the DNS client to use a DNS server on your network.

### Before you begin

- Ensure that you have a domain name server on your network.

### Procedure

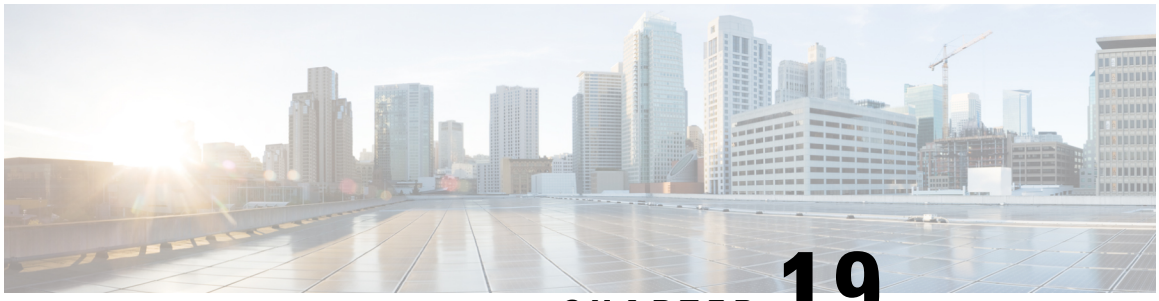
| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configuration terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# vrf context managment | Specifies a configurable virtual and routing (VRF) name. |
| **Step 3** | switch(config)# **{ip \| ipv6} host** *name ipv/ipv6 address1 [ip/ipv6 address2... ip/ipv6 address6]* | Defines up to six static hostname-to-address mappings in the host name cache. |
| **Step 4** | (Optional) switch(config)# **ip domain name** *name* [**use-vrf** *vrf-name*] | Defines the default domain name server that Cisco NX-OS uses to complete unqualified |

| | Command or Action | Purpose |
|---|---|---|
| | | hostnames. You can optionally define a VRF that Cisco NX-OS uses to resolve this domain name server if it cannot be resolved in the VRF that you configured this domain name under.<br>Cisco NX-OS appends the default domain name to any host name that does not contain a complete domain name before starting a domain-name lookup. |
| **Step 5** | (Optional) switch(config)# **ip domain-list** *name* [**use-vrf** *vrf-name*] | Defines additional domain name servers that Cisco NX-OS can use to complete unqualified hostnames. You can optionally define a VRF that Cisco NX-OS uses to resolve this domain name server if it cannot be resolved in the VRF that you configured this domain name under.<br>Cisco NX-OS uses each entry in the domain list to append that domain name to any hostname that does not contain a complete domain name before starting a domain-name lookup. Cisco NX-OS continues this for each entry in the domain list until it finds a match. |
| **Step 6** | (Optional) switch(config)# **ip name-server** *ip/ipv6 server-address1 [ip/ipv6 server-address2... ip/ipv6 server-address6]* **[use-vrf** *vrf-name*] | Defines up to six name servers. The address can be either an IPv4 address or an IPv6 address.<br>You can optionally define a VRF that Cisco NX-OS uses to reach this name server if it cannot be reached in the VRF that you configured this name server under. |
| **Step 7** | (Optional) switch(config)# **ip domain-lookup** | Enables DNS-based address translation. This feature is enabled by default. |
| **Step 8** | (Optional) switch(config)# **show hosts** | Displays information about DNS. |
| **Step 9** | switch(config)# **exit** | Exits configuration mode and returns to EXEC mode. |
| **Step 10** | (Optional) switch# **copy running-config startup-config** | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to configure a default domain name and enable DNS lookup:

```
switch# config t
switch(config)# vrf context management
switch(config)# ip domain-name mycompany.com
switch(config)# ip name-server 172.68.0.10
switch(config)# ip domain-lookup
```



CHAPTER 19

# Configuring sFlow

This chapter contains the following sections:


- Information About sFlow, on page 219
- Prerequisites, on page 220
- Guidelines and Limitations for sFlow, on page 220
- Default Settings for sFlow, on page 220
- Configuring sFLow, on page 220
- Verifying the sFlow Configuration, on page 227
- Configuration Examples for sFlow, on page 227
- Additional References for sFlow, on page 227
- Feature History for sFlow, on page 228


# Information About sFlow

sFlow allows you to monitor the real-time traffic in data networks that contain switches and routers. It uses the sampling mechanism in the sFlow Agent software on switches and routers for monitoring traffic and to forward the sample data on ingress and egress ports to the central data collector, also called the sFlow Analyzer.

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 in the Cisco NX-OS software, 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 the information about the ingress port, egress port, and the original packet length. An sFlow datagram can have multiple sFlow samples.

# Prerequisites

You must enable the sFlow feature using the **feature sflow** command to configure sFlow.

# Guidelines and Limitations for sFlow

The sFlow configuration guidelines and limitations are as follows:

- 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.
- sFlow egress sampling for multicast, broadcast, or unknown unicast packets is not supported.
- You should configure the sampling rate based on the sFlow configuration and traffic in the system.
- Cisco Nexus 3000 Series supports only one sFlow collector.

# Default Settings for sFlow

*Table 31: Default sFlow Parameters*

| Parameters | Default |
|---|---|
| sFlow sampling-rate | 4096 |
| sFlow sampling-size | 128 |
| sFlow max datagram-size | 1400 |
| sFlow collector-port | 6343 |
| sFlow counter-poll-interval | 20 |

# Configuring sFLow

## Enabling the sFlow Feature

You must enable the sFlow feature before you can configure sFlow on the switch.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **feature sflow** | Enables the sFlow feature. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | (Optional) **show feature** | Displays enabled and disabled features. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to enable the sFlow feature:

```
switch# configure terminal
switch(config)# feature sflow
switch(config)# copy running-config startup-config
```

# Configuring the Sampling Rate

### Before you begin

Ensure that you have enabled the sFlow feature.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **sflow sampling-rate** *sampling-rate* | Configures the sFlow sampling rate for packets.<br>The *sampling-rate* can be an integer between 4096-1000000000. The default value is 4096. |
| **Step 3** | (Optional) **show sflow** | Displays sFlow information. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to set the sampling rate to 50,000:

```
switch# configure terminal
switch(config)# sflow sampling-rate 50000
switch(config)# copy running-config startup-config
```

# Configuring the Maximum Sampled Size

You can configure the maximum number of bytes that should be copied from a sampled packet.

### Before you begin

Ensure that you have enabled the sFlow feature.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **sflow max-sampled-size** *sampling-size* | Configures the sFlow maximum sampling size packets.<br>The range for the *sampling-size* is from 64 to 256 bytes. The default value is 128. |
| **Step 3** | (Optional) **show sflow** | Displays sFlow information. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure the maximum sampling size for the sFlow Agent:

```
switch# configure terminal
switch(config)# sflow max-sampled-size 200
switch(config)# copy running-config startup-config
```

# 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

Ensure that you have enabled the sFlow feature.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **sflow counter-poll-interval** *poll-interval* | Configures the sFlow poll interval for an interface. The range for the *poll-interval* is from 0 to 2147483647 seconds. The default value is 20. |
| **Step 3** | (Optional) **show sflow** | Displays sFlow information. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure the sFlow poll interval for an interface:

```
switch# configure terminal
switch(config)# sflow counter-poll-interval 100
switch(config)# copy running-config startup-config
```

# 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

Ensure that you have enabled the sFlow feature.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **sflow max-datagram-size** *datagram-size* | Configures the sFlow maximum datagram size.<br>The range for the *datagram-size* is from 200 to 9000 bytes. The default value is 1400. |
| **Step 3** | (Optional) **show sflow** | Displays sFlow information. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure the sFlow maximum datagram size:

```
switch# configure terminal
switch(config)# sflow max-datagram-size 2000
switch(config)# copy running-config startup-config
[########################################] 100%
```

# Configuring the sFlow Analyzer Address

### Before you begin

Ensure that you have enabled the sFlow feature.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **sflow collector-ip** *IP-address vrf-instance* | Configures the IPv4 address for the sFlow Analyzer.<br>*vrf-instance* can be one of the following:<br>• A user-defined VRF name—You can specify a maximum of 32 alphanumeric characters.<br>• **vrf management**— You must use this option if the sFlow data collector is on the network connected to the management port.<br>• **vrf default**— You must use this option if the sFlow data collector is on the network connected to the front panel ports. |
| **Step 3** | (Optional) **show sflow** | Displays sFlow information. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure the IPv4 address of the sFlow data collector that is connected to the management port:

```
switch# configure terminal
switch(config)# sflow collector-ip 192.0.2.5 vrf management
switch(config)# copy running-config startup-config
```

# Configuring the sFlow Analyzer Port

You can configure the destination port for sFlow datagrams.

### Before you begin

Ensure that you have enabled the sFlow feature.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **sflow collector-port** *collector-port* | Configures the UDP port of the sFlow Analyzer.<br>The range for the *collector-port* is from 0 to 65535. The default value is 6343. |
| **Step 3** | (Optional) **show sflow** | Displays sFlow information. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure the destination port for sFlow datagrams:

```
switch# configure terminal
switch(config)# sflow collector-port 7000
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

# Configuring the sFlow Agent Address

### Before you begin

Ensure that you have enabled the sFlow feature.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | [**no**] **sflow agent-ip** *ip-address* | Configures the IPv4 address of the sFlow Agent.<br>The default *ip-address* is 0.0.0.0, which means that all sampling is disabled on the switch. You must specify a valid IP address to enable sFlow functionality.<br>**Note**<br>This IP address is not necessarily the source IP address for sending the sFlow datagram to the collector. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | (Optional) **show sflow** | Displays sFlow information. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure the IPv4 address of the sFlow Agent:

```
switch# configure terminal
switch(config)# sflow agent-ip 192.0.2.3
switch(config)# copy running-config startup-config
```

# Configuring the sFlow Sampling Data Source

The sFlow sampling data source can be an Ethernet port, a range of Ethernet ports, or a port channel.

### Before you begin

- Ensure that you have enabled the sFlow feature.
- If you want to use a port channel as the data source, ensure that you have already configured the port channel and you know the port channel number.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# [**no**] **sflow data-source interface** [**ethernet** *slot*/*port*[*-port*] \|**port-channel** *channel-number*] | Configures the sFlow sampling data source.<br>For an Ethernet data source, *slot* is the slot number and *port* can be either a single port number or a range of ports designated as *port-port*. |
| **Step 3** | (Optional) switch(config)# **show sflow** | Displays sFlow information. |
| **Step 4** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to configure Ethernet ports 5 through 12 for the sFlow sampler:

```
switch# configure terminal
switch(config)# sflow data-source interface ethernet 1/5-12
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

This example shows how to configure port channel 100 for the sFlow sampler:

```
switch# configure terminal
switch(config)# sflow data-source interface port-channel 100
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

# Verifying the sFlow Configuration

Use the following commands to verify the sFlow configuration information:

| Command | Purpose |
|---|---|
| **show sflow** | Displays the sFlow global configuration. |
| **show sflow statistics** | Displays the sFlow statistics. |
| **clear sflow statistics** | Clears the sFlow statistics. |
| **show running-config sflow** [**all**] | Displays the current running sFlow configuration. |

# Configuration Examples for sFlow

This example shows how to configure sFlow:

```
feature sflow
sflow sampling-rate 5000
sflow max-sampled-size 200
sflow counter-poll-interval 100
sflow max-datagram-size 2000
sflow collector-ip 192.0.2.5 vrf management
sflow collector-port 7000
sflow agent-ip 192.0.2.3
sflow data-source interface ethernet 1/5
```

# Additional References for sFlow

***Table 32: Related Documents for sFlow***

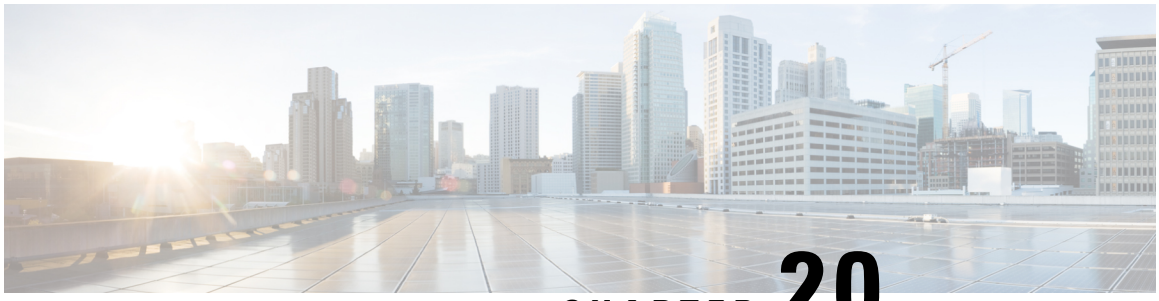
| Related Topic | Document Title |
|---|---|
| sFlow CLI commands | *Cisco Nexus 3000 Series NX-OS System Management Command Reference.* |

| Related Topic | Document Title |
|---|---|
| RFC 3176 | Defines the sFlow packet format and SNMP MIB.<br>http://www.sflow.org/rfc3176.txt |

# Feature History for sFlow

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

| Feature Name | Releases | Feature Information |
|---|---|---|
| sFlow | 5.0(3)U4(1) | This feature was introduced. |



CHAPTER 20

# Configuring Tap Aggregation and MPLS Stripping

This chapter contains the following sections:


- Information About Tap Aggregation, on page 229
- Information About MPLS Stripping, on page 231
- Configuring Tap Aggregation, on page 232
- Verifying the Tap Aggregation Configuration, on page 235
- Configuring MPLS Stripping, on page 236
- Verifying the MPLS Label Configuration, on page 239


# Information About Tap Aggregation

## Network Taps

You can use various methods to monitor packets. One method uses physical hardware 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, it is desirable for a third party to monitor 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:

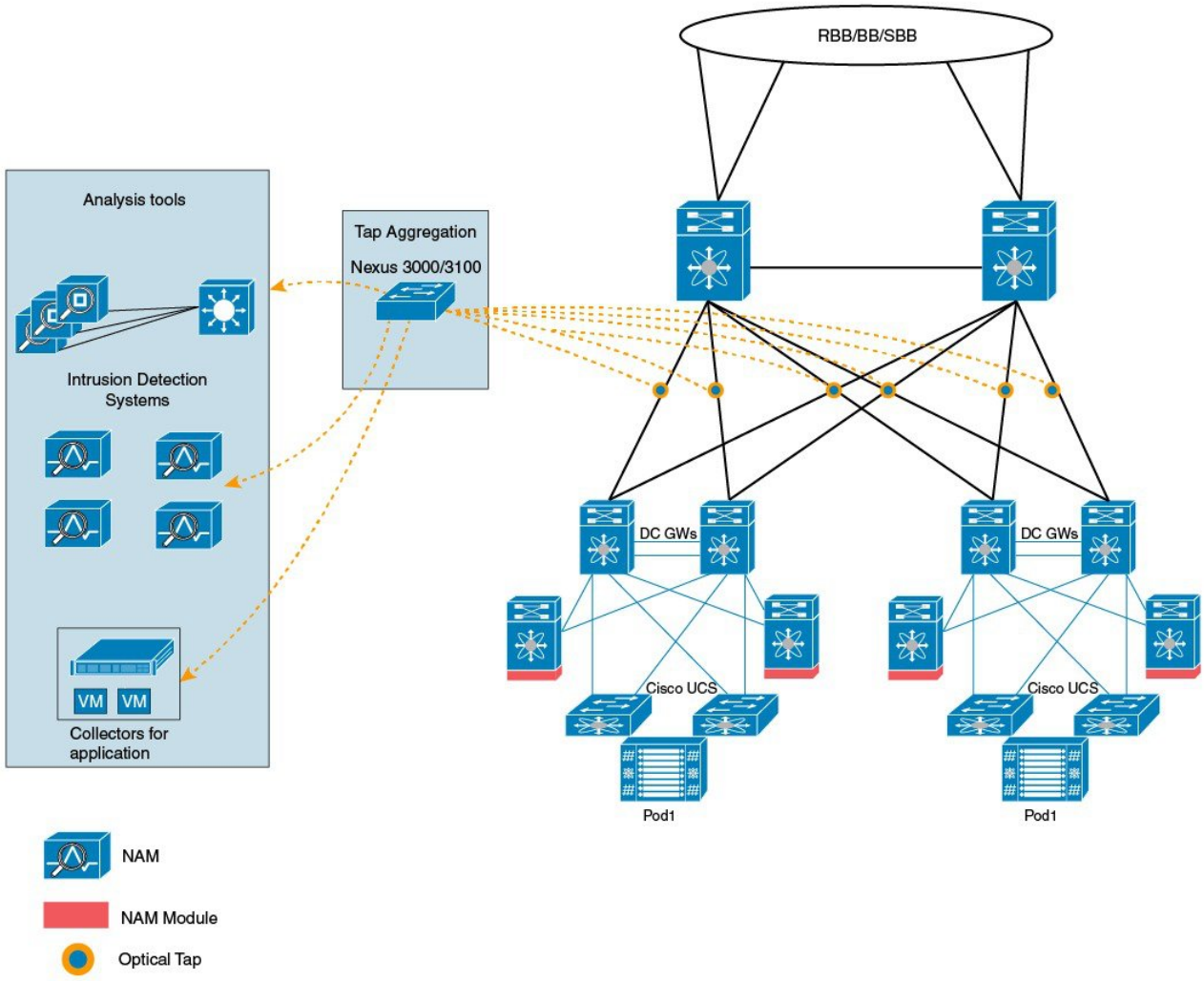
- They can handle full-duplex data transmission
- They are nonobtrusive and not detectable by the network with no physical or logical addressing
- Some taps support full inline power with the capability to build a distributed tap

Whether 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

An alternative solution to help with monitoring and troubleshooting tasks in the data center is a device that is especially designated to allow the aggregation of multiple taps and that also connects to multiple monitoring systems. This solution is referred to as tap aggregation. Tap aggregation switches link all the monitoring devices directly to specific points in the network fabric that handle the packets that need to be observed.

***Figure 2: Tap Aggregation Switch Solution***




In the tap aggregation switch solution, the Cisco Nexus 3000 or Cisco Nexus 3100 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 itself is directly connected to all 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 dynamically program the tap aggregation switch with a configuration that allows traffic to enter the switch through a certain set of ports that are connected to the network elements. You can also configure a number of match criteria and actions to filter specific traffic and redirect them to one or more tools.

## Guidelines and Limitations for Tap Aggregation

Tap aggregation has the following guidelines and limitations:

- The interface to be applied with the tap aggregation policy must be in Layer 2. You can configure a Layer 3 interface with the policy, but the policy becomes nonfunctional.
- Each rule must be associated with only one unique match criterion.
- All tap aggregation interfaces must share the same ACL. Multiple ACLs are not required across interfaces because the match criteria includes an ingress interface.
- The actions **vlan-set** and **vlan-strip** must always be specified after the **redirect** action. Otherwise, the entry will be rejected as invalid.
- The deny rule does not support actions such as **redirect**, **vlan-set**, and **vlan-strip**.
- When you enter a list of inputs, for example, a list of interfaces for the 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, ensure that you enter the whole interface type and not just the short form of it. For example, ensure that you enter ethernet1/1 instead of eth1/1 and port-channel 50 instead of po50.

# Information About MPLS Stripping

## MPLS Overview

Multiprotocol Label Switching (MPLS) integrates the performance and traffic management capabilities of Layer 2 switching with the scalability, flexibility, and performance of Layer 3 routing.

An MPLS architecture provides the following benefits:

- Data can be transferred over any combination of Layer 2 technologies
- Support is offered for all Layer 3 protocols
- Scaling is possible well beyond anything offered in today's networks

## MPLS Header Stripping

The ingress ports of Cisco Nexus 3172 receive various 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 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

Because the MPLS label is imposed between the Layer 2 header and the Layer 3 header, its headers and data are not located at the standard byte offset. Standard network monitoring tools cannot monitor and analyze this traffic. To enable standard network monitoring tools to monitor this traffic, single-labeled packets are stripped off their MPLS label headers and redirected to T-cache devices.

MPLS packets with multiple label headers are sent to deep packet inspection (DPI) devices without stripping their MPLS headers.

## Guidelines and Limitations for MPLS Stripping

MPLS stripping has the following guidelines and limitations:

- Disable all Layer 3 and vPC features before you enable MPLS stripping.
- Ensure that global tap-aggregation mode is enabled.
- The ingress and egress interfaces involved in MPLS stripping must have **mode 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.
- Only one tap ACL is supported on the system.
- 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.
- To enable MPLS stripping, ensure that you configure the Control Plane Policing (CoPP) class for MPLS, copp-s-mpls.
- For MPLS stripped packets, port-channel load balancing is supported.
- 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 VLAN is also stripped with the MPLS label.
- MPLS stripping is supported only on Cisco Nexus 3100 Series switches.

# Configuring Tap Aggregation

## Enabling Tap Aggregation

Ensure that you run the **copy running-config startup-config** command and reload the switch after enabling tap aggregation.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch (config)# **[no] hardware profile tap-aggregation [l2drop]** | Enables tap aggregation and reserves entries in the interface table that are needed for VLAN tagging.<br>The **l2drop** option drops non-IP traffic ingress on tap interfaces.<br>The **no** form of this command disables the feature. |
| **Step 3** | switch (config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |
| **Step 4** | switch (config)# **reload** | Reloads the Cisco NX-OS software. |

#### Example

This example shows how to configure tap aggregation globally on the switch:

```
switch# configure terminal
switch(config)# hardware profile tap-aggregation
switch(config)# copy running-config startup-config
switch(config)# reload
```

# Configuring a Tap Aggregation Policy

You can configure a TAP aggregation policy on an IP access control list (ACL) or on a MAC ACL.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | • switch(config)# **ip access-list** *access-list-name*<br>• switch(config)# **mac access-list** *access-list-name* | Creates an IP ACL and enters IP access list configuration mode or creates a MAC ACL and enters MAC access list configuration mode.<br>**Note**<br>Starting with Release 7.0(3)I5(1), support for IPv6 ACLs is added on the Cisco Nexus 3000 Series switches. The redirect action is supported in IPv6 ACLs. All the match options that are currently supported for IPv6 PACL are now supported with the redirect action. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | switch(config-acl)# **statistics per-entry** | Starts recording statistics for how many packets are permitted or denied by each entry. |
| **Step 4** | switch(config-acl)# **[no] permit** *protocol source destination match-criteria action* | Creates an IP access control list (ACL) rule that permits traffic to match its conditions.<br>The **no** version of this command removes the permit rule from the policy.<br>*match-criteria* can be one of the following:<br>• **ingress-intf**<br>**Note** The ingress interface can be a match criteria only on Layer 2—EtherType or port channel<br>• **vlan**<br>• **vlan-priority**<br>**Note** Each policy can have only one rule associated with a unique match criterion.<br>*action* can be one of the following:<br>• **redirect**<br>• **priority**<br>• **set-vlan**<br>A tap ACL that matches on non-IP ethertype must be specified with a priority value greater than 0. |
| **Step 5** | switch(config-acl)# **[no] deny** *protocol source destination match-criteria action* | Creates an IP access control list (ACL) rule that denies traffic matching its conditions.<br>The **no** version of this command removes the deny rule from the policy.<br>It does not support **redirect**, and **vlan-set** actions. |

### Example

This example shows how to configure a tap aggregation policy:

```
switch# configure terminal
switch(config)# ip access-list test
switch(config-acl)# statistics per-entry
switch(config-acl)# permit ip any any ingress-intf Ethernet1/4 redirect Ethernet1/8
switch(config-acl)# permit ip any any ingress-intf Ethernet1/6 redirect
```

```
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
switch(config-acl)# permit tcp any eq www any ingress-intf Ethernet1/10 redirect port-channel4
switch(config-acl)# deny ip any any
```

# Attaching a Tap Aggregation Policy to an Interface

To attach a tap aggregation policy to an interface, enter the tap aggregation mode and apply the ACL configured with tap aggregation to the interface. Ensure that the interface to which you attach the policy is a Layer 2 interface.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **interface** *type slot/port* | Enters the interface configuration mode for the specified interface. |
| **Step 3** | switch (config-if)# **[no] mode tap-aggregation** | Allows an attachment of the ACL with the match and action criteria.<br>The **no** form of this command disallows the attachment of an ACL with the tap aggregation policy to the interface. To remove the ACL from the interface, use the **no ip port access-group** command. |
| **Step 4** | switch(config-if)# **[no] ip port access-group** *access-list-name* **in** | Applies an IPv4 access control list (ACL) to an interface as a port ACL.<br>The **no** form of this command removes an ACL from an interface. |

### Example

This example shows how to attach a tap aggregation policy to an interface:

```
switch# configure terminal
switch(config)# interface ethernet1/2
switch (config-if)# mode tap-aggregation
switch(config-if)# ip port access-group test in
```

# Verifying the Tap Aggregation Configuration

| Command | Purpose |
|---|---|
| **show ip access-list** *access-list-name* | Displays all IPv4 access control lists (ACLs) or a specific IPv4 ACL. |

### Example

This example shows how to display an IPv4 ACL:

```
switch(config)# show ip access-list test
IPV4 ACL test
        10 permit ip any any ethertype 0x800 ingress-intf Ethernet1/4 redirect E
thernet1/8
        20 permit ip any any ingress-intf Ethernet1/6 redirect Ethernet1/1,Ether
net1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
        30 permit tcp any eq www any ethertype 0x800 ingress-intf Ethernet1/10 r
edirect port-channel4
        40 deny ip any any
```

# Configuring MPLS Stripping

## Enabling MPLS Stripping

You can enable MPLS stripping globally.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **[no] mpls strip** | Globally enables MPLS stripping.<br>The **no** form of this command disables MPLS stripping. |

### Example

The following example shows how to enable MPLS stripping:

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

## Adding and Deleting MPLS Labels

The device can learn the labels dynamically whenever a frame is received with an unknown label on a mode tap interface. You can also add or delete static MPLS labels by using the following commands:

### Before you begin

- Enable tap aggregation
- Configure tap aggregation policy
- Attach a tap aggregation policy to an interface

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **mpls strip label** *label* | Adds the specified static MPLS label.<br>The value of the label can range from 1 to 1048575. |
| **Step 3** | switch(config)# **no mpls strip label** *label* \| **all** | Deletes the specified static MPLS label.<br>The **all** option deletes all static MPLS labels. |

### Example

The following 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
```

The following example shows how to delete a static MPLS label:

```
switch# configure terminal
switch(config)# no mpls strip label 200
```

The following example shows how to delete all static MPLS labels:

```
switch# configure terminal
switch(config)# no mpls strip label all
```

# Clearing Label Entries

You can clear dynamic label entries from the MPLS label table by using the following command:

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **clear mpls strip label dynamic** | Clears dynamic label entries from the MPLS label table. |

### Example

The following example shows how to clear dynamic label entries:

```
switch# clear mpls strip label dynamic
```

# Clearing MPLS Stripping Counters

You can clear all software and hardware MPLS stripping counters.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **clear counters mpls strip** | Clears all MPLS stripping counters. |

### Example

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
```

# 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** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **mpls strip label-age** *age* | Specifies the amount of time after which dynamic MPLS labels age out. |

### Example

The following example shows how to configure label age for dynamic MPLS labels:

```
switch# configure terminal
switch(config)# mpls strip label-age 300
```

## Configuring Destination MAC Addresses

You can configure the destination MAC address for stripped egress frames.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config)# **mpls strip dest-mac** *mac-address* | Specifies the destination MAC address for egress frames that are stripped of their headers.<br>The MAC address can be specified in one of the following four formats:<br>• E.E.E<br>• EE-EE-EE-EE-EE-EE<br>• EE:EE:EE:EE:EE:EE<br>• EEEE.EEEE.EEEE |

### Example

The following example shows how to configure the destination MAC address for egress frames:

```
switch# configure terminal
switch(config)# mpls strip dest-mac 1.1.1
```

# Verifying the MPLS Label Configuration

Use the following command to display the MPLS label configuration:

| Command | Purpose |
|---|---|
| **show mpls strip labels** [*label* \| **all** \| **dynamic** \|**static**] | Displays information about MPLS labels. You can specify the following options:<br>• *label*—Label to be displayed<br>• **all**—Specifies that all labels must be displayed. This is the default option.<br>• **dynamic**—Specifies that only dynamic labels must be displayed.<br>• **static**—Specifies that only static labels must be displayed. |

### Example

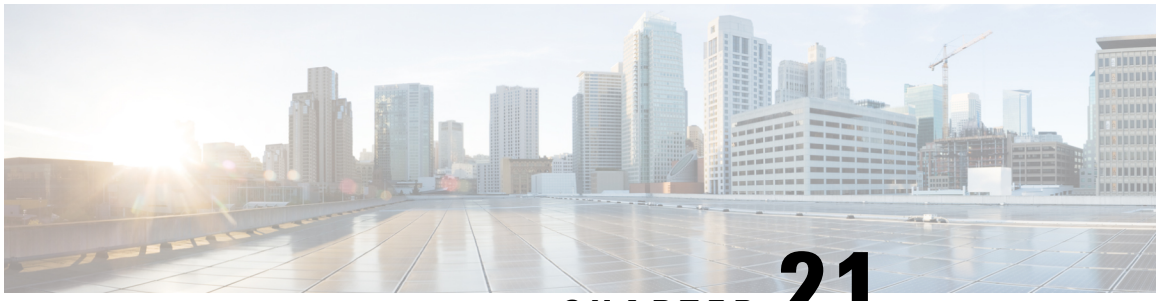
The following example shows how to display all MPLS labels:

```
switch# show mpls strip labels
MPLS Strip Labels:
    Total      : 3005
    Static     : 5
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    Eth1/53/1            15            1              210
     4097    Eth1/53/1            15            1              210
     4098    Eth1/53/1            15            1              210
     4099    Eth1/53/1             7            2              219
     4100    Eth1/53/1             7            2              219
     4101    Eth1/53/1             7            2              219
     4102    Eth1/53/1            39            1              206
     4103    Eth1/53/1            39            1              206
     4104    Eth1/53/1            39            1              206
     4105    Eth1/53/1             1            1              217
     4106    Eth1/53/1             1            1              217
     4107    Eth1/53/1             1            1              217
     4108    Eth1/53/1            15            1              210
*   25000    None <User>          39            1              206
*   20000    None <User>          39            1              206
*   21000    None <User>           1            1              217
```

The following example shows how to display only static MPLS labels:

```
switch(config)# show mpls strip labels static
MPLS Strip Labels:
    Total      : 3005
    Static     : 5
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
```

```
--------------------------------------------------------------------------------
*      300    None <User>          403            0                0
*      100    None <User>          416            0                0
*    25000    None <User>          869            0                0
*    20000    None <User>          869            0                0
*    21000    None <User>          869            0                0
```



CHAPTER 21

# Configuring Graceful Insertion and Removal

This chapter describes how to configure graceful insertion and removal (GIR) on the Cisco Nexus 3000 Series switches.


- About Graceful Insertion and Removal, on page 243
- GIR Workflow, on page 245
- Configuring the Maintenance-Mode Profile, on page 246
- Configuring the Normal-Mode Profile, on page 247
- Creating a Snapshot, on page 249
- Adding Show Commands to Snapshots, on page 250
- Triggering Graceful Removal, on page 252
- Triggering Graceful Insertion, on page 255
- Maintenance Mode Enhancements, on page 256
- Verifying the GIR Configuration, on page 257


## 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.

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** *instance-number seconds* | Delays the execution of the command by a specified number of seconds. You can delay multiple instances of the command.<br>The range for the *instance-number* and *seconds* arguments is from 0 to 2177483647. |
| **python instance** *instance-number uri* [*python-arguments*]<br>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.<br>You can enter a maximum of 32 alphanumeric characters for the Python arguments. |

## 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

# 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 246.)
2. (Optional) Create the normal-mode profile. (See Configuring the Normal-Mode Profile, on page 247.)
3. Take a snapshot before triggering graceful removal. (See Creating a Snapshot, on page 249.)
4. Trigger graceful removal to put the switch in maintenance mode. (See Triggering Graceful Removal, on page 252.)
5. Trigger graceful insertion to return the switch to normal mode. (See Triggering Graceful Insertion, on page 255.)
6. Take a snapshot after triggering graceful insertion. (See Creating a Snapshot, on page 249.)

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 257.)

# 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**<br>**Example:**<br>`switch# configure maintenance profile maintenance-mode`<br>`Enter configuration commands, one per line. End with CNTL/Z.`<br>`switch(config-mm-profile)#` | Enters a configuration session for the maintenance-mode profile. The **no** option deletes the maintenance profile maintenance-mode.<br>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 244. |
| **Step 2** | **end**<br>**Example:**<br>`switch(config-mm-profile)# end`<br>`switch#` | Closes the maintenance-mode profile. |
| **Step 3** | **show maintenance profile maintenance-mode**<br>**Example:**<br>`switch# show maintenance profile maintenance-mode` | 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**<br>**Example:**<br>`switch# configure maintenance profile normal-mode`<br>`Enter configuration commands, one per line. End with CNTL/Z.`<br>`switch(config-mm-profile)#` | Enters a configuration session for the normal-mode profile. The **no** version removes the maintenance profile normal-mode.<br>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 244. |
| **Step 2** | **end**<br>**Example:**<br>`switch(config-mm-profile)# end`<br>`switch#` | Closes the normal-mode profile. |
| **Step 3** | **show maintenance profile normal-mode**<br>**Example:**<br>`switch# show maintenance profile normal-mode` | 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** *snapshot-name description*<br>**Example:**<br>`switch# snapshot create snap_before_maintenance Taken before maintenance`<br>`Executing 'show interface'... Done`<br>`Executing 'show ip route summary vrf all'... Done`<br>`Executing 'show ipv6 route summary vrf all'... Done`<br>`Executing 'show bgp sessions vrf all'... Done`<br>`Executing 'show ip eigrp topology summary'... Done`<br>`Executing 'show ipv6 eigrp topology summary'... Done`<br>`Feature 'vpc' not enabled, skipping...`<br>`Executing 'show ip ospf vrf all'... Done`<br>`Feature 'ospfv3' not enabled, skipping...`<br>`Feature 'isis' not enabled, skipping...`<br>`Feature 'rip' not enabled, skipping...`<br>`Snapshot 'snap_before_maintenance' created` | Captures the running state or operational data of selected features and stores the data on persistent storage media.<br>You can enter a maximum of 64 alphanumeric chapters for the snapshot name and a maximum of 254 alphanumeric characters for the description.<br>Use the **snapshot delete** {**all** \| *snapshot-name*} command to delete all snapshots or a specific snapshot. |
| **Step 2** | **show snapshots**<br>**Example:**<br>`switch# show snapshots`<br>`Snapshot Name Time Description`<br>`---`<br>`snap_before_maintenance Wed Aug 19 13:53:28 2015 Taken before maintenance` | Displays snapshots present on the switch. |
| **Step 3** | **show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary** \| **ipv4routes** \| **ipv6routes**]<br>**Example:**<br>`switch# show snapshots compare snap_before_maintenance snap_after_maintenance` | Displays a comparison of two snapshots.<br>The **summary** option displays just enough information to see the overall changes between the two snapshots.<br>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*]<br>**Example:**<br>`switch# snapshot section add myshow "show ip interface brief" ROW_intf intf-name` | Adds a user-specified section to snapshots. The *section* 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.<br>The *row-id* argument specifies the tag of each row entry of the **show** command's XML output. The *element-key1* and *element-key2* arguments specify the tags used to distinguish among row entries. In most cases, only the *element-key1* argument needs to specified to be able to distinguish among row entries.<br>**Note**<br>To delete a user-specified section from snapshots, use the **snapshot section delete** *section* command. |
| **Step 2** | **show snapshots sections**<br>**Example:**<br>`switch# show snapshots sections` | Displays the user-specified snapshot sections. |
| **Step 3** | **show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary** \| **ipv4routes** \| **ipv6routes**]<br>**Example:**<br>`switch# show snapshots compare snap1 snap2` | Displays a comparison of two snapshots.<br>The **summary** option displays just enough information to see the overall changes between the two snapshots.<br>The **ipv4routes** and **ipv6routes** options display the changes in IPv4 and IPv6 routes between the two snapshots. |

### Example

The following example adds the **show ip interface brief** command to the myshow snapshot section. It also compares two snapshots (snap1 and snap2) and shows the user-specified sections in both snapshots.

```
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

switch# show snapshots compare snap1 snap2
```

```
================================================================================
Feature              Tag                  snap1                snap2
================================================================================

[bgp]
--------------------------------------------------------------------------------
.............

[interface]
--------------------------------------------------------------------------------

       [interface:mgmt0]
                     vdc_lvl_in_pkts      692310               **692317**
                     vdc_lvl_in_mcast     575281               **575287**
                     vdc_lvl_in_bcast     77209                **77210**
                     vdc_lvl_in_bytes     63293252             **63293714**
                     vdc_lvl_out_pkts     41197                **41198**
                     vdc_lvl_out_ucast    33966                **33967**
                     vdc_lvl_out_bytes    6419714              **6419788**
.............

[ospf]
--------------------------------------------------------------------------------
.............

[myshow]
--------------------------------------------------------------------------------

       [interface:Ethernet1/1]
                     state                up                   **down**
                     admin_state          up                   **down**
.............
```

# 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 246.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **system mode maintenance** [**dont-generate-profile** \| **timeout** *value* \| **shutdown** \| **on-reload reset-reason** *reason*] | Puts all enabled protocols in maintenance mode (using the **isolate** command). |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`switch(config)# system mode maintenance`<br>`Following configuration will be applied:`<br><br>`ip pim isolate`<br>`router bgp 65502`<br>`isolate`<br>`router ospf p1`<br>`isolate`<br>`router ospfv3 p1`<br>`isolate`<br><br>`Do you want to continue (y/n)? [no] y`<br><br>`Generating a snapshot before going into maintenance mode`<br><br>`Starting to apply commands...`<br><br>`Applying : ip pim isolate`<br>`Applying : router bgp 65502`<br>`Applying : isolate`<br>`Applying : router ospf p1`<br>`Applying : isolate`<br>`Applying : router ospfv3 p1`<br>`Applying : isolate`<br><br>`Maintenance mode operation successful.` | The following options are available:<br>• **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.<br>• **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.<br>• **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.<br>• **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.<br>The maintenance mode reset reasons are as follows:<br>• HW_ERROR—Hardware error<br>• SVC_FAILURE—Critical service failure<br>• KERN_FAILURE—Kernel panic<br>• WDOG_TIMEOUT—Watchdog timeout<br>• FATAL_ERROR—Fatal error<br>• LC_FAILURE—Line card failure<br>• MATCH_ANY—Any of the above reasons |

| | Command or Action | Purpose |
|---|---|---|
| | | The system prompts you to continue. Enter **y** to continue or **n** to terminate the process. |
| **Step 3** | (Optional) **show system mode**<br>**Example:**<br>`switch(config)# show system mode`<br>`System Mode: Maintenance` | Displays the current system mode.<br>The switch is in maintenance mode. You can now perform any desired debugging or upgrade operations on the switch. |
| **Step 4** | (Optional) **copy running-config startup-config**<br>**Example:**<br>`switch(config)# copy running-config startup-config` | 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 246.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure terminal**<br>**Example:**<br>`switch# configure terminal`<br>`switch(config)#` | Enters global configuration mode. |
| **Step 2** | **no system mode maintenance** [**dont-generate-profile**]<br>**Example:**<br>`switch(config)# no system mode maintenance dont-generate-profile`<br>`Following configuration will be applied:`<br><br>`no ip pim isolate`<br>`router bgp 65502`<br>`no isolate`<br>`router ospf p1`<br>`no isolate`<br>`router ospfv3 p1`<br>`no isolate`<br><br>`Do you want to continue (y/n)? [no] y`<br><br>`Starting to apply commands...`<br><br>`Applying : no ip pim isolate`<br>`Applying : router bgp 65502`<br>`Applying : no isolate`<br>`Applying : router ospf p1`<br>`Applying : no isolate`<br>`Applying : router ospfv3 p1`<br>`Applying : no isolate`<br><br>`Maintenance mode operation successful.`<br><br>`Generating Current Snapshot` | Puts all enabled protocols in normal mode (using the **no isolate** command).<br>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.<br>The system prompts you to continue. Enter **y** to continue or **n** to terminate the process. |
| **Step 3** | (Optional) **show system mode**<br>**Example:** | Displays the current system mode. The switch is now in normal mode and is fully operational. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch(config)# show system mode`<br>`System Mode: Normal` | |

# Maintenance Mode Enhancements

Starting with Release 7.0(3)I5(1), the following maintenance mode enhancements have been added to Cisco Nexus 3000 Series switches:

- In the system maintenance shutdown mode, the following message is added:

  ```
  NOTE: The command system interface shutdown will shutdown all interfaces excluding mgmt
   0.
  ```

- 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 under #ifdef for Cisco Nexus 9000 Series switches only.

  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(m-mode)#`.

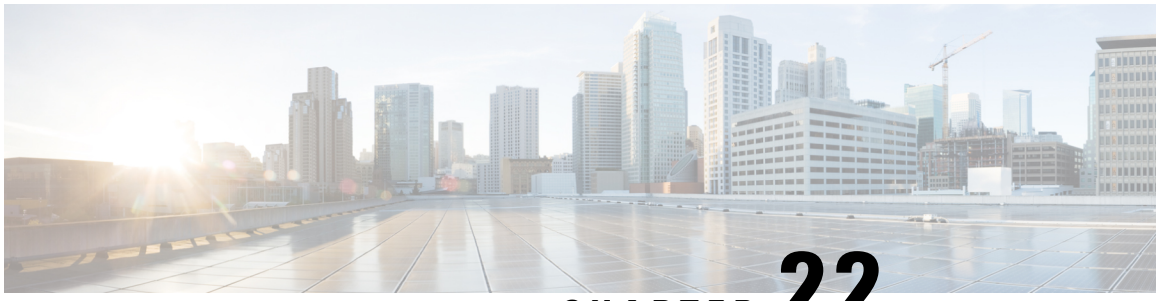
- Support for the SNMP traps has been added when the device moves from the maintenance mode to the normal mode and vice-versa through CLI reload, or system reset. The **snmp-server enable traps mmode**

**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 252. |
| **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** *snapshot-name-1 snapshot-name-2* [**summary** \| **ipv4routes** \| **ipv6routes**] | Displays a comparison of two snapshots.<br>The **summary** option displays just enough information to see the overall changes between the two snapshots.<br>The **ipv4routes** and **ipv6routes** options display the changes in IPv4 and IPv6 routes between the two snapshots. |
| **show snapshots dump** *snapshot-name* | 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. |



CHAPTER 22

# Performing Software Maintenance Upgrades (SMUs)

This chapter describes how to perform software maintenance upgrades (SMUs) on Cisco Nexus 3000 Series switches.

This chapter includes the following sections:


- About SMUs, on page 259
- Prerequisites for SMUs, on page 260
- Guidelines and Limitations for SMUs, on page 261
- Performing a Software Maintenance Upgrade for Cisco NX-OS, on page 261


## 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 immediate issues. All defects fixed by SMUs are integrated into the maintenance releases.

For information on upgrading your device to a new feature or maintenance release, see the *Cisco Nexus 3000 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.

**Note** Beginning with Cisco NX-OS Release 7.0(3)I2(1), SMU package files have an .rpm extension. Earlier files have a .bin extension.

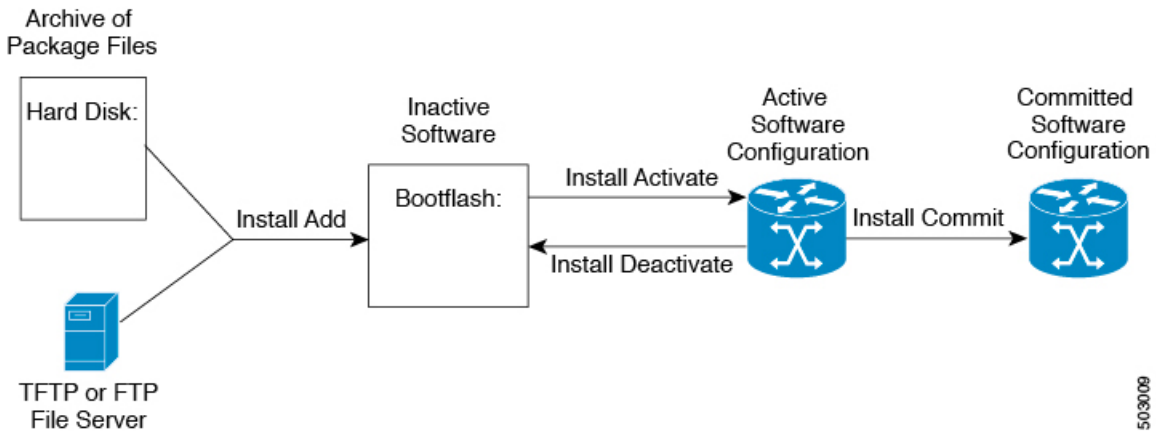
# Package Management

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 3: Process to Add, Activate, and Commit SMU Packages***




# 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:

- 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.
- The package being activated must be compatible with the current active software set.
- You cannot activate multiple SMUs in one command.
- Activation is performed only after the package compatibility checks have been passed. If a conflict is found, an error message displays.
- 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.
- If you perform a software maintenance upgrade and later upgrade your device to a new Cisco Nexus 3000 software release, the new image will overwrite both the previous Cisco Nexus 3000 release and the SMU package file.

# 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 stable and prepared for the software changes.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **show install active**<br>**Example:**<br>`switch# show install active` | 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. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | **show module**<br>**Example:**<br>`switch# show module` | Confirms that all modules are in the stable state. |
| **Step 3** | **show clock**<br>**Example:**<br>`switch# show clock` | Verifies that the system clock is correct. Software operations use certificates based on device clock times. |

### Example

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
Active Packages:
Active Packages on Module #3:

Active Packages on Module #6:

Active Packages on Module #7:
Active Packages on Module #22:

Active Packages on Module #30:
```

This example shows how to display the current system clock setting:

```
switch# show clock
02:14:51.474 PST Wed Jan 04 2014
```

# 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.

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** This procedure uses Cisco NX-OS CLI commands to add and activate RPM package files. If you would prefer to use YUM commands, follow the instructions in the "Installing RPMs from Bash" section of the Cisco Nexus 3000 Series NX-OS Programmability Guide.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **install add** *filename* [**activate**]<br>**Example:**<br>`switch# install add bootflash: nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm` | 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.<br>The *filename* argument can take any of these formats:<br>• **bootflash:***filename*<br>• **tftp:**//*hostname-or-ipaddress/directory-path/filename*<br>• **ftp:**//*username*:*password*@*hostname-or-ipaddress/directory-path/filename*<br>• **sftp:**//*hostname-or-ipaddress/directory-path/filename* |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | (Optional) **show install inactive**<br>**Example:**<br>`switch# show install inactive` | Displays the inactive packages on the device. Verify that the package added in the previous step appears in the display. |
| **Step 3** | Required: **install activate** *filename* [**test**]<br>**Example:**<br>`switch# install activate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm`<br>**Example:**<br>`switch# install activate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm Install operation 1 completed successfully at Wed Mar 16 00:42:12 2016`<br>**Example:**<br>`switch# install activate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm Install operation 2 !!WARNING!! This patch will get activated only after a reload of the switch. at Wed Mar 16 00:42:12 2016` | Activates a package that was added to the device. SMU packages remain inactive until activated. (Skip this step if the package was activated earlier with the **install add activate** command.)<br>**Note**<br>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** | Repeat Step 3 until all packages are activated. | Activates additional packages as required. |
| **Step 5** | (Optional) **show install active**<br>**Example:**<br>`switch# show install active` | 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.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **install commit** *filename*<br>**Example:**<br>`switch# install commit nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm` | Commits the current set of packages so that these packages are used if the device is restarted. |
| **Step 2** | (Optional) **show install committed**<br>**Example:** | Displays which packages are committed. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch# show install committed` | |

# 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.

**Note** This procedure uses Cisco NX-OS CLI commands to deactivate and remove RPM package files. If you would prefer to use YUM commands, follow the instructions in the "Erasing an RPM" section of the Cisco Nexus 3000 Series NX-OS Programmability Guide.

## Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **install deactivate** *filename*<br>**Example:**<br>`switch# install deactivate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm` | Deactivates a package that was added to the device and turns off the package features for the line card.<br>**Note** Press **?** after a partial package name to display all possible matches available for deactivation. If there is only one match, press the **Tab** key to fill in the rest of the package name. |
| **Step 2** | (Optional) **show install inactive**<br>**Example:**<br>`switch# show install inactive` | Displays the inactive packages on the device. |
| **Step 3** | (Optional) **install commit**<br>**Example:**<br>`switch# install commit` | Commits the current set of packages so that these packages are used if the device is restarted.<br>**Note** Packages can be removed only if the deactivation operation is committed. |
| **Step 4** | (Optional) **install remove** {*filename* \| **inactive**}<br>**Example:**<br>`switch# install remove nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm`<br>`Proceed with removing nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm? (y/n)? [n] y`<br>**Example:** | Removes the inactive package.<br>• Only inactive packages can be removed.<br>• Packages can be removed only if they are deactivated from all line cards in the device.<br>• The package deactivation must be committed. |

| | Command or Action | Purpose |
|---|---|---|
| | `switch# install remove inactive`<br>`Proceed with removing? (y/n)? [n] y` | • To remove a specific inactive package from a storage device, use the **install remove** command with the *filename* argument.<br>• To remove all inactive packages from all nodes in the system, use the **install remove** command with the **inactive** keyword. |

# Downgrading Feature RPMs

Follow this procedure to downgrade an installed feature RPM to the base feature RPM.

**Note** This procedure uses Cisco NX-OS CLI commands to downgrade feature RPMs. If you would prefer to use YUM commands, follow the instructions in the "Downgrading an RPM" section of the Cisco Nexus 3000 Series NX-OS Programmability Guide.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | (Optional) **show install packages**<br>**Example:**<br>`switch# show install packages`<br>`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**<br>**Example:**<br>`switch# run bash`<br>`bash-4.2$` | Loads Bash. |
| **Step 3** | Required: **ls ****feature******<br>**Example:**<br>`bash-4.2$ ls *ntp*`<br>`ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm` | Lists the RPM for the specified feature. |
| **Step 4** | Required: **cp** *filename* **/bootflash**<br>**Example:**<br>`bash-4.2$ cp`<br>`ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm`<br>`/bootflash` | Copies the base feature RPM to the bootflash. |
| **Step 5** | Required: **exit**<br>**Example:** | Exits Bash. |

| | Command or Action | Purpose |
|---|---|---|
| | `bash-4.2$ exit` | |
| **Step 6** | Required: **install add bootflash:***filename* **activate downgrade**<br>**Example:**<br>`switch# install add bootflash:ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm activate downgrade`<br>`Adding the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) [############ ] 60%`<br>`Adding the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) [####################] 100%`<br>`Install operation 11 completed successfully at Thu Sep 8 15:35:35 2015`<br>`Activating the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm)`<br>`This install operation requires system reload. Do you wish to continue (y/n)?: [n] y`<br>`[ 217.975959] [1473348971] writing reset reason 132, System reset due to reload patch(es) activation`<br>`[ 217.991166] [1473348971]\ufffd\ufffd CISCO SWITCH Ver7.51`<br>`Device detected on 0:6:0 after 0 msecs`<br>`Device detected on 0:1:1 after 0 msecs`<br>`Device detected on 0:1:0 after 0 msecs`<br>`MCFrequency 1333Mhz`<br>`Relocated to memory` | Downgrades the feature RPM.<br>**Note**<br>If you are prompted to reload the device, enter **Y**. A reload is required only when downgrading the NTP and SNMP feature RPMs. |
| **Step 7** | (Optional) **show install packages \| i** *feature*<br>**Example:**<br>`switch# show install packages \| i ntp`<br>`ntp.lib32_n9000 1.0.0-7.0.3.I2.2e installed` | Displays the base feature RPM on the device. |

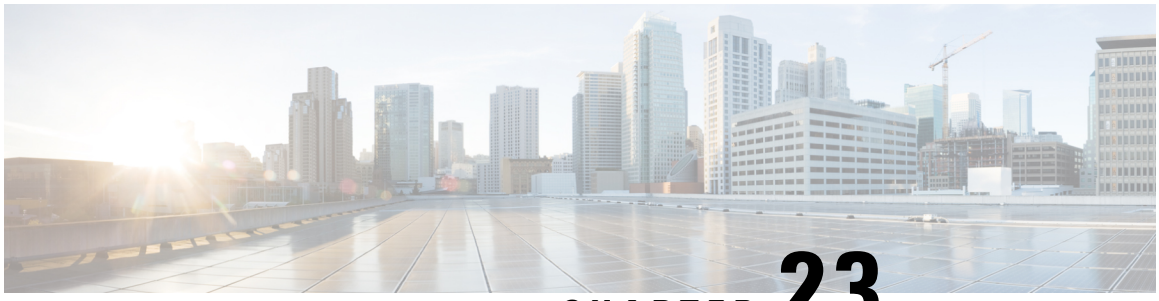
# Displaying Installation Log Information

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:

```
switch# show install log
Wed Mar 16 01:26:09 2016
Install operation 1 by user 'admin' at Wed Mar 16 01:19:19 2016
Install add bootflash: nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 1 completed successfully at Wed Mar 16 01:19:24 2016
----------------------------------------
Install operation 2 by user 'admin' at Wed Mar 16 01:19:29 2016
Install activate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 2 completed successfully at Wed Mar 16 01:19:45 2016
----------------------------------------
Install operation 3 by user 'admin' at Wed Mar 16 01:20:05 2016
Install commit nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 3 completed successfully at Wed Mar 16 01:20:08 2016
----------------------------------------
Install operation 4 by user 'admin' at Wed Mar 16 01:20:21 2016
Install deactivate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 4 completed successfully at Wed Mar 16 01:20:36 2016
----------------------------------------
Install operation 5 by user 'admin' at Wed Mar 16 01:20:43 2016
Install commit nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 5 completed successfully at Wed Mar 16 01:20:46 2016
----------------------------------------
Install operation 6 by user 'admin' at Wed Mar 16 01:20:55 2016
Install remove nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 6 completed successfully at Wed Mar 16 01:20:57 2016
```



CHAPTER 23

# Performing Configuration Replace

This chapter includes the following sections:


- About Configuration Replace and Commit-timeout, on page 269
- Overview, on page 269
- Guidelines and Limitations for Configuration Replace, on page 271
- Recommended Workflow for Configuration Replace, on page 272
- Performing a Configuration Replace, on page 272
- Verifying Configuration Replace , on page 274
- Examples for Configuration Replace, on page 275


## 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** <*target-url*> 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** <*source-url*> **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** <*target-url*> command. | You can use a partial configuration file as a source file for the **copy** <*source-url*> **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.
- 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.
- 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:

- The configuration replace feature does not support features that require a reload. One such feature is: system vlan reserve.
- The configuration replace feature is not supported on Cisco Nexus 9500 platform switches with -R line cards.
- 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.
- 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.

# Recommended Workflow for Configuration Replace

The following workflow is the recommended workflow for configuration replace:

# 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** { < *uri_local* > \| < *uri_remote* > } [ **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. |
| **Step 2** | **configure replace** [ **bootflash** / **scp** / **sftp** ] < *user-configuration-file* > **show-patch** | Displays the differences between the running-configuration and the user-provided configuration. |
| **Step 3** | **configure replace** [ **bootflash** / **scp** / **sftp** ] < *user-configuration-file* > **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** <*user-configuration-file*> [**best-effort**] | Replaces the configuration on the switch with the new user configuration and enables the configuration replace with semantic validation.<br>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.<br>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** <*user-configuration-file*> [**verify-and-commit**] | Replaces the configuration on the switch with the new user configuration and enables the configuration replace with semantic validation.<br>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.<br>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.<br>**Note** This step is applicable only if you have configured the commit-timeout feature.<br>**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*> *non-interactive* | 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. |

# Verifying Configuration Replace

To check and verify configuration replace and its status, use the commands that are outlined in the table:

***Table 33: 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. |

| Command | Purpose |
|---|---|
| **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 parallelly 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 sematic 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
------------------------------------------------------------------------------
```

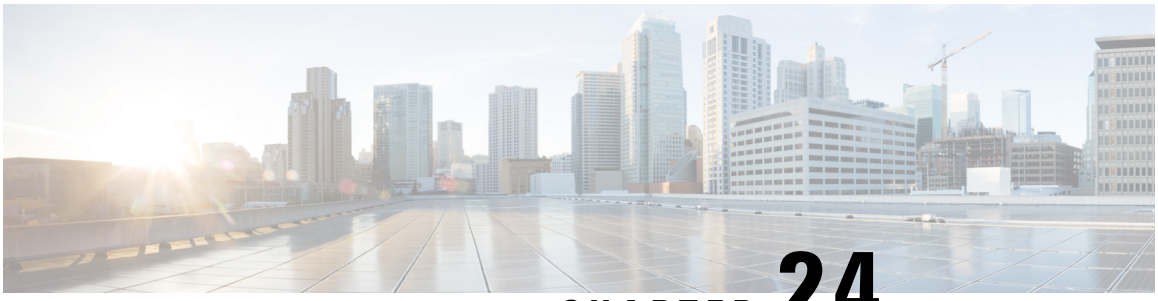
```
time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS
End Time             : Wed, 06:39:47 25 Jan 2017
Rollback Status      : Success

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)#
```



CHAPTER 24

# Configuring Rollback

This chapter contains the following sections:


- Information About Rollbacks, on page 279
- Guidelines and Limitations for Rollbacks, on page 279
- Creating a Checkpoint, on page 280
- Implementing a Rollback, on page 281
- Verifying the Rollback Configuration, on page 282


## Information About Rollbacks

The rollback feature allows you to take a snapshot, or user checkpoint, of the Cisco NX-OS configuration and then reapply that configuration to your switch at any point without having to reload the switch. A rollback allows any authorized administrator to apply this checkpoint configuration without requiring expert knowledge of the features configured in the checkpoint.

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 an atomic rollback. An atomic rollback implements a rollback only if no errors occur.

## Guidelines and Limitations for Rollbacks

A rollback has the following configuration guidelines and limitations:

- You can create up to ten checkpoint copies.
- You cannot apply the checkpoint file of one switch into another switch.
- Your checkpoint file names must be 75 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 you enter the **write erase** and **reload** command, checkpoints are deleted. You can use the clear checkpoint database command to clear out all checkpoint files.
- 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 are local to a switch.
- Checkpoints that are created using the **checkpoint** and **checkpoint** *checkpoint_name* commands are present upon a switchover for all switches.
- 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
- 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 per switch.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **checkpoint** { [*cp-name*] [**description** *descr*] \|**file** *file-name*<br>**Example:**<br>`switch# checkpoint stable` | 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 user-checkpoint-<number> where number is from 1 to 10.<br>The description can contain up to 80 alphanumeric characters, including spaces. |
| **Step 2** | (Optional) switch# **no checkpoint***cp-name*<br>**Example:**<br>`switch# no checkpoint stable` | You can use the **no** form of the **checkpoint** command to remove a checkpoint name.<br>Use the **delete** command to remove a checkpoint file. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | (Optional) switch# **show checkpoint***cp-name*<br>**Example:**<br>[ **all**]<br>`switch# show checkpoint stable` | 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** *src-cp-name* \| **running-config** \| **startup-config** \| **file** *source-file*} {**checkpoint** *dest-cp-name* \| **running-config** \| **startup-config** \| **file** *dest-file*}<br>**Example:**<br>`switch# show diff rollback-patch checkpoint stable running-config` | Displays the differences between the source and destination checkpoint selections. |
| **Step 2** | **rollback running-config** {**checkpoint** *cp-name* \| **file** *cp-file*} **atomic**<br>**Example:**<br>`switch# rollback running-config checkpoint stable` | Creates an atomic rollback to the specified checkpoint name or file if no errors occur. |

### Example

The following example shows how to create a checkpoint file and then implement an atomic rollback to a user checkpoint name:

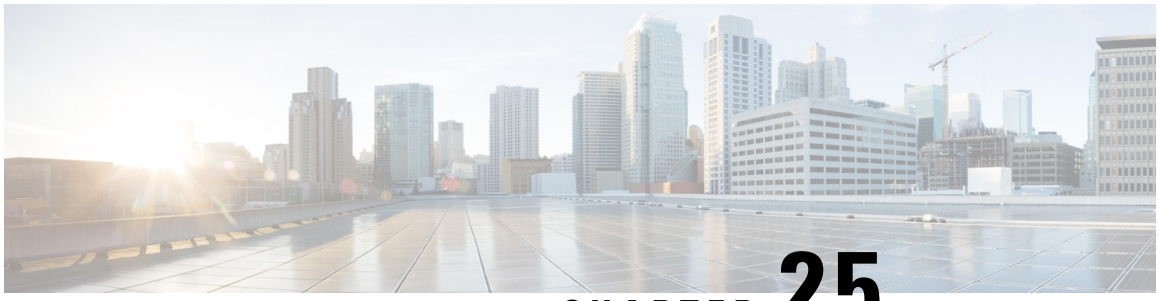
```
switch# checkpoint stable
switch# rollback running-config checkpoint stable atomic
```

# Verifying the Rollback Configuration

Use the following commands to verify the rollback configuration:

| Command | Purpose |
|---|---|
| **show checkpoint** *name* [ **all**] | Displays the contents of the checkpoint name. |
| **show checkpoint all** [**user** \| **system**] | Displays the contents of all checkpoints in the current switch. You can limit the displayed checkpoints to user or system-generated checkpoints. |
| **show checkpoint summary** [**user** \| **system**] | Displays a list of all checkpoints in the current switch. You can limit the displayed checkpoints to user or system-generated checkpoints. |
| **show diff rollback-patch** {**checkpoint** *src-cp-name* \| **running-config** \| **startup-config** \| **file** *source-file*} {**checkpoint** *dest-cp-name* \| **running-config** \| **startup-config** \| **file** *dest-file*} | Displays the differences between the source and destination checkpoint selections. |
| **show rollback log** [**exec** \| **verify**] | Displays the contents of the rollback log. |

**Note** Use the **clear checkpoint database** command to delete all checkpoint files.



CHAPTER 25

# Configuring User Accounts and RBAC

This chapter contains the following sections:


- Information About User Accounts and RBAC, on page 283
- Guidelines and Limitations for User Accounts, on page 286
- Configuring User Accounts, on page 287
- Configuring RBAC, on page 289
- Verifying the User Accounts and RBAC Configuration, on page 293
- Configuring User Accounts Default Settings for the User Accounts and RBAC, on page 294


# Information About User Accounts and RBAC

Cisco Nexus Series switches use role-based access control (RBAC) to define the amount of access that each user has when the user logs into the switch.

With RBAC, you define one or more user roles and then specify which management operations each user role is allowed to perform. When you create a user account for the switch, you associate that account with a user role, which then determines what the individual user is allowed to do on the switch.

## User Roles

User roles contain rules that define the operations allowed for the user who is assigned the role. Each user role can contain multiple rules and each user can have multiple roles. For example, if role1 allows access only to configuration operations, and role2 allows access only to debug operations, users who belong to both role1 and role2 can access configuration and debug operations. You can also limit access to specific VLANs, and interfaces.

The switch provides the following default user roles:

**network-admin (superuser)**

Complete read and write access to the entire switch.

**network-operator**

Complete read access to the switch.

**Note** If you belong to multiple roles, you can execute a combination of all the commands permitted by these roles. Access to a command takes priority over being denied access to a command. For example, suppose a user has RoleA, which denied access to the configuration commands. However, the user also has RoleB, which has access to the configuration commands. In this case, the user has access to the configuration commands.

# Rules

The rule is the basic element of a role. A rule defines what operations the role allows the user to perform. You can apply rules for the following parameters:

**Command**

A command or group of commands defined in a regular expression.

**Feature**

Commands that apply to a function provided by the Cisco Nexus device. Enter the **show role feature** command to display the feature names available for this parameter.

**Feature group**

Default or user-defined group of features. Enter the **show role feature-group** command to display the default feature groups available for this parameter.

**OID**

An SNMP object identifier (OID).

These parameters create a hierarchical relationship. The most basic control parameter is the command. The next control parameter is the feature, which represents all commands associated with the feature. The last control parameter is the feature group. The feature group combines related features and allows you to easily manage the rules.

SNMP OID is supported for RBAC. You can configure a read-only or read-and-write rule for an SNMP OID.

You can configure up to 256 rules for each role. The user-specified rule number determines the order in which the rules are applied. Rules are applied in descending order. For example, if a role has three rules, rule 3 is applied before rule 2, which is applied before rule 1.

# User Role Policies

You can define user role policies to limit the switch resources that the user can access, or to limit access to interfaces and VLANs.

User role policies are constrained by the rules defined for the role. For example, if you define an interface policy to permit access to specific interfaces, the user does not have access to the interfaces unless you configure a command rule for the role to permit the **interface** command.

If a command rule permits access to specific resources (interfaces, VLANs), the user is permitted to access these resources, even if the user is not listed in the user role policies associated with that user.

# User Account Configuration Restrictions

The following words are reserved and cannot be used to configure users:

- adm
- bin
- daemon
- ftp
- ftpuser
- games
- gdm
- gopher
- halt
- lp
- mail
- mailnull
- man
- mtsuser
- news
- nobody
- san-admin
- shutdown
- sync
- sys
- uucp
- xfs

**Caution** The Cisco Nexus Series switch does not support all numeric usernames, even if those usernames were created in TACACS+ or RADIUS. If an all numeric username exists on an AAA server and is entered during login, the switch rejects the login request.

# User Password Requirements

Cisco Nexus device passwords are case sensitive and can contain alphanumeric characters. Special characters, such as the dollar sign ($) or the percent sign (%), are not allowed.

**Note** Beginning with Cisco NX-OS Release 7.2(0)N1(1), special characters, such as the dollar sign ($) or the percent sign (%), can be used in Cisco Nexus device passwords.

If a password is trivial (such as a short, easy-to-decipher password), the Cisco Nexus device rejects the password. Be sure to configure a strong password for each user account. A strong password has the following characteristics:

- At least eight characters long
- Does not contain many consecutive characters (such as "abcd")
- Does not contain many repeating characters (such as "aaabbb")
- Does not contain dictionary words
- Does not contain proper names
- Contains both uppercase and lowercase characters
- Contains numbers

The following are examples of strong passwords:

- If2CoM18
- 2009AsdfLkj30
- Cb1955S21

**Note** For security reasons, user passwords do not display in the configuration files.

# Guidelines and Limitations for User Accounts

User accounts have the following guidelines and limitations when configuring user accounts and RBAC:

- Regardless of the read-write rule configured for a user role, some commands can be executed only through the predefined network-admin role.
- Starting with Release 7.0(3)I2(1), a new criteria is implemented to check the password strength.
- Up to 256 rules can be added to a user role.
- A maximum of 64 user roles can be assigned to a user account.
- You can assign a user role to more that one user account.
- Predefined roles such as network-admin, network-operator, and san-admin are not editable.
- Add, delete, and editing of rules is not supported for the SAN admin user role.
- The interface, VLAN, and/or VSAN scope cannot be changed for the SAN admin user role.

**Note** A user account must have at least one user role.

# Configuring User Accounts

**Note** Changes to user account attributes do not take effect until the user logs in and creates a new session.

You can use any alphanumeric character (or) an _ (underscore) as the first character in a username. Using any other special characters for the first character is not allowed. If the username contains the characters that are not allowed, the specified user is unable to log in.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | (Optional) switch(config)# **show role** | Displays the user roles available. You can configure other user roles, if necessary. |
| **Step 3** | switch(config) # **username** *user-id* [**password** *password*] [**expire** *date*] [**role** *role-name*] | Configures a user account.<br>The *user-id* is a case-sensitive, alphanumeric character string with a maximum of 28 characters.<br>The default *password* is undefined.<br>**Note**<br>If you do not specify a password, the user might not be able to log into the switch.<br>**Note**<br>Starting with Release 7.0(3)I2(1), a new internal function is implemented to check the password strength. When enabling the password strength-check on Cisco Nexus 3000 Series platforms in Release 7.0(3)I2(1), it has a different criteria than the previous releases.<br>The **expire** *date* option format is YYYY-MM-DD. The default is no expiry date. |
| **Step 4** | switch(config) # **exit** | Exists global configuration mode. |
| **Step 5** | (Optional) switch# **show user-account** | Displays the role configuration. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | (Optional) switch# **copy running-config startup-config** | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to configure a user account:

```
switch# configure terminal
switch(config)# username NewUser password 4Ty18Rnt
switch(config)# exit
switch# show user-account
```

The following example shows the criteria in enabling the password strength-check starting with Release 7.0(3)I2(1):

```
switch(config)# username xyz password nbv12345
password is weak
Password should contain characters from at least three of the following classes: lower case
 letters, upper case letters, digits and special characters.
switch(config)# username xyz password Nbv12345
password is weak
it is too simplistic/systematic
switch(config)#
```

# Configuring SAN Admin Users

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **username** *user-id* **role san-admin password** *password* | Configures SAN admin user role access for the specified user. |
| **Step 3** | (Optional) switch(config) # **show user-account** | Displays the role configuration. |
| **Step 4** | (Optional) switch(config) # **show snmp-user** | Displays the SNMP user configuration. |
| **Step 5** | (Optional) switch(config)# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

The following example shows how to configure a SAN admin user and display the user account and SNMP user configuration:

```
switch# configure terminal
switch(config)# username user1 role san-admin password xyz123
switch(config)# show user-account
```

```
user:admin
     this user account has no expiry date
     roles:network-admin
user:user1
     this user account has no expiry date
     roles:san-admin
switch(config) # show snmp user
_____________________________________________________________________________
     SNMP USERS
_____________________________________________________________________________

User     Auth  Priv(enforce)  Groups
____     ____  _____________  ______
admin    md5   des(no)        network-admin
user1    md5   des(no)        san-admin

_____________________________________________________________________________
NOTIFICATION TARGET USES (configured for sending V3 Inform)
_____________________________________________________________________________

User     Auth   Priv
____     ____   ____

switch(config) #
```

# Configuring RBAC

## Creating User Roles and Rules

The rule number that you specify determines the order in which the rules are applied. Rules are applied in descending order. For example, if a role has three rules, rule 3 is applied before rule 2, which is applied before rule 1.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **role name** *role-name* | Specifies a user role and enters role configuration mode.<br>The *role-name* argument is a case-sensitive, alphanumeric character string with a maximum of 16 characters. |
| **Step 3** | switch(config-role) # **rule** *number* {**deny** \| **permit**} **command** *command-string* | Configures a command rule.<br>The *command-string* can contain spaces and regular expressions. For example, interface ethernet * includes all Ethernet interfaces.<br>Repeat this command for as many rules as needed. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | switch(config-role)# **rule** *number* {**deny** \| **permit**} {**read** \| **read-write**} | Configures a read-only or read-and-write rule for all operations. |
| **Step 5** | switch(config-role)# **rule** *number* {**deny** \| **permit**} {**read** \| **read-write**} **feature** *feature-name* | Configures a read-only or read-and-write rule for a feature.<br>Use the **show role feature** command to display a list of features.<br>Repeat this command for as many rules as needed. |
| **Step 6** | switch(config-role)# **rule** *number* {**deny** \| **permit**} {**read** \| **read-write**} **feature-group** *group-name* | Configures a read-only or read-and-write rule for a feature group.<br>Use the **show role feature-group** command to display a list of feature groups.<br>Repeat this command for as many rules as needed. |
| **Step 7** | (Optional) switch(config-role)# **description** *text* | Configures the role description. You can include spaces in the description. |
| **Step 8** | switch(config-role)# **end** | Exits role configuration mode. |
| **Step 9** | (Optional) switch# **show role** | Displays the user role configuration. |
| **Step 10** | (Optional) switch# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to create user roles and specify rules:

```
switch# configure terminal
switch(config)# role name UserA
switch(config-role)# rule deny command clear users
switch(config-role)# rule deny read-write
switch(config-role)# description This role does not allow users to use clear commands
switch(config-role)# end
switch(config)# show role
```

# Creating Feature Groups

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | switch(config) # **role feature-group** *group-name* | Specifies a user role feature group and enters role feature group configuration mode.<br>The *group-name* is a case-sensitive, alphanumeric character string with a maximum of 32 characters. |
| **Step 3** | switch(config) # **exit** | Exits global configuration mode. |
| **Step 4** | (Optional) switch# **show role feature-group** | Displays the role feature group configuration. |
| **Step 5** | (Optional) switch# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

### Example

This example shows how to create a feature group:

```
switch# configure terminal
switch(config) # role feature-group group1
switch(config) # exit
switch# show role feature-group
switch# copy running-config startup-config
switch#
```

# Changing User Role Interface Policies

You can change a user role interface policy to limit the interfaces that the user can access. Specify a list of interfaces that the role can access. You can specify it for as many interfaces as needed.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **role name** *role-name* | Specifies a user role and enters role configuration mode. |
| **Step 3** | switch(config-role) # **interface policy deny** | Enters role interface policy configuration mode. |
| **Step 4** | switch(config-role-interface) # **permit interface** *interface-list* | Specifies a list of interfaces that the role can access.<br>Repeat this command for as many interfaces as needed.<br>For this command, you can specify Ethernet interfaces. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 5** | switch(config-role-interface) # **exit** | Exits role interface policy configuration mode. |
| **Step 6** | (Optional) switch(config-role) # **show role** | Displays the role configuration. |
| **Step 7** | (Optional) switch(config-role) # **copy running-config startup-config** | Copies the running configuration to the startup configuration. |

### Example

The following example shows how to change a user role interface policy to limit the interfaces that the user can access:

```
switch# configure terminal
switch(config)# role name UserB
switch(config-role)# interface policy deny
switch(config-role-interface)# permit interface ethernet 2/1
switch(config-role-interface)# permit interface fc 3/1
switch(config-role-interface)# permit interface vfc 30/1
```

# Changing User Role VLAN Policies

You can change a user role VLAN policy to limit the VLANs that the user can access.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config) # **role name** *role-name* | Specifies a user role and enters role configuration mode. |
| **Step 3** | switch(config-role )# **vlan policy deny** | Enters role VLAN policy configuration mode. |
| **Step 4** | switch(config-role-vlan # **permit vlan** *vlan-list* | Specifies a range of VLANs that the role can access.<br>Repeat this command for as many VLANs as needed. |
| **Step 5** | switch(config-role-vlan) # **exit** | Exits role VLAN policy configuration mode. |
| **Step 6** | (Optional) switch# **show role** | Displays the role configuration. |
| **Step 7** | (Optional) switch# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

# Changing User Role VSAN Policies

You can change a user role VSAN policy to limit the VSANs that the user can access.

### Procedure

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | switch# **configure terminal** | Enters global configuration mode. |
| **Step 2** | switch(config-role) # **role name** *role-name* | Specifies a user role and enters role configuration mode. |
| **Step 3** | switch(config-role) # **vsan policy deny** | Enters role VSAN policy configuration mode. |
| **Step 4** | switch(config-role-vsan) # **permit vsan** *vsan-list* | Specifies a range of VSANs that the role can access.<br>Repeat this command for as many VSANs as needed. |
| **Step 5** | switch(config-role-vsan) # **exit** | Exits role VSAN policy configuration mode. |
| **Step 6** | (Optional) switch# **show role** | Displays the role configuration. |
| **Step 7** | (Optional) switch# **copy running-config startup-config** | Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration. |

# Verifying the User Accounts and RBAC Configuration

Use one of the following commands to verify the configuration:

| Command | Purpose |
|---|---|
| **show role** [*role-name*] | Displays the user role configuration |
| **show role feature** | Displays the feature list. |
| **show role feature-group** | Displays the feature group configuration. |
| **show startup-config security** | Displays the user account configuration in the startup configuration. |
| **show running-config security** [**all**] | Displays the user account configuration in the running configuration. The **all** keyword displays the default values for the user accounts. |
| **show user-account** | Displays user account information. |

# Configuring User Accounts Default Settings for the User Accounts and RBAC

The following table lists the default settings for user accounts and RBAC parameters.

*Table 34: Default User Accounts and RBAC Parameters*

| Parameters | Default |
|---|---|
| User account password | Undefined. |
| User account expiry date | None. |
| Interface policy | All interfaces are accessible. |
| VLAN policy | All VLANs are accessible. |
| VFC policy | All VFCs are accessible. |
| VETH policy | All VETHs are accessible. |

# INDEX

## A


ACL log 82
  match level 82
ACL logging 80
  applying to an interface 80
ACL logging cache 79
  configuring 79
action statements 165
  EEM 165
action statements, configuring 172
  EEM 172
activating sessions 188
  SPAN 188
adding MPLS labels 236
adding show commands, alert groups 103
  smart call home 103
additional references 179
  EEM 179
agent address 225
  sFlow 225
alert groups 91
  smart call home 91
analyzer address 224
  sFlow 224
analyzer port 224
  sFlow 224
associating alert groups 103
  smart call home 103
Attaching a Tap Aggregation policy to an interface 235


## C


cache 79
  logging 79
    configuring 79
call home notifications 109
  full-text format for syslog 109
  XML format for syslog 109
clearing label entries 237
clearing MPLS counters 238
committing 67
  NTP configuration changes 67
configuration example 211, 227
  ERSPAN 211
    source 211
  sFlow 227
configuration examples 69, 190
  for SPAN 190
  NTP 69
configuration sync after reboot 22
  switch profiles 22
configuration, verifying 125
  scheduler 125
configure maintenance profile maintenance-mode 246
configure maintenance profile normal-mode 248
configuring 57–59, 61, 63, 66
  device as an authoritative NTP server 57
  NTP authentication 59, 61
  NTP logging 66
  NTP server and peer 58
  NTP source interface 63
  NTP source IP address 63
Configuring a Tap Aggregation Policy 233
configuring destination MAC address 239
Configuring MPLS aging 238
contact information, configuring 99
  smart call home 99
counter poll interval 222
  sFlow 222
creating, deleting sessions 184
  SPAN 184


## D


datagram size 223
  sFlow 223
default parameters 199
  ERSPAN 199
default settings 56, 98, 116, 118, 167, 220
  EEM 167
  rollback 116
  scheduler 118
  sFlow 220
  smart call home 98
default SNMP settings 131
defining EEM policies 174
  VSH script 174

deleting MPLS labels 236
description, configuring 187
  SPAN 187
destination ports, characteristics 182
  SPAN 182
destination profile, creating 100
  smart call home 100
destination profile, modifying 101
  smart call home 101
destination profiles 90
  smart call home 90
destinations 182
  SPAN 182
device IDs 93
  call home format 93
diagnostics 155–157, 159
  configuring 157
  default settings 159
  expansion modules 157
  health monitoring 156
  runtime 155
disabling 124
  scheduler 124
discarding 68
  NTP configuration changes 68
displaying information 189
  SPAN 189
displaying installation log information 267
downgrading software 183, 195
  loss of ERSPAN configurations 195
  loss of SPAN configurations 183
duplicate message throttling, disabling 106–107
  smart call home 106–107

## E

e-mail details, configuring 104
  smart call home 104
e-mail notifications 89
  smart call home 89
EEE 166
  guidelines and limitations 166
EEM 164–169, 172, 174–176, 179
  action statements 165
  action statements, configuring 172
  additional references 179
  default settings 167
  defining environment variables 167
  event statements 164
  event statements, configuring 169
  feature history 179
  licensing 166
  policies 164
  prerequisites 166
  syslog script 176
EEM *(continued)*
  system policies,overriding 175
  user policy, defining 168
  VSH script 174
    registering and activating 174
  VSH script policies 166
egress frames, configuring destination MAC addresses 239
embedded event manager 163
  overview 163
enabling 67, 119
  CFS distribution for NTP 67
  scheduler 119
enabling MPLS stripping 236
Enabling Tap Aggregation 232
environment variables, defining 167
  EEM 167
ERSPAN 193–195, 199, 211, 213
  configuration loss when downgrading software 195
  configuring source sessions 199
  default parameters 199
  destinations 194
  guidelines and limitations 195
  high availability 195
  information about 193
  prerequisites 195
  related documents 213
  sessions 195
    multiple 195
  source 211
    configuration example 211
  source sessions 199
    configuring for ERSPAN 199
  sources 193
Ethernet destination port, configuring 185
  SPAN 185
event statements 164
  EEM 164
event statements, configuring 169
  EEM 169
example 125–126
  job schedule, displaying 126
  scheduler job, creating 125
  scheduler job, scheduling 125
  scheduler jobs, displaying results 126
example, local and peer sync 29
  switch profiles 29
executing a session 115

## F

facility messages logging 77
  configuring 77
feature groups, creating 290
  RBAC 290
feature history 179, 228
  EEM 179

feature history *(continued)*
sFlow 228
filtering SNMP requests 134

## G

GOLD diagnostics 155–157
configuring 157
expansion modules 157
health monitoring 156
runtime 155
guidelines 195, 220
ERSPAN 195
sFlow 220
guidelines and limitations 10, 43, 55, 72, 97, 118, 131, 166, 183, 231, 286
EEM 166
for NTP 55
PTP 43
scheduler 118
smart call home 97
SNMP 131
SPAN 183
switch profiles 10
system message logging 72
user accounts 286
guidelines and limitations for configuration rollback 279
guidelines and limitations for MPLS stripping 232

## H

header stripping 231
health monitoring diagnostics 156
information 156
high availability 43
PTP 43
high availability 43

## I

IDs 93
serial IDs 93
information 53
ntp 53
information about 54, 117
clock manager 54
distributing NTP using CFS 54
NTP as time server 54
scheduler 117
interfaces, configuring 46
PTP 46
isolate 244

## J

job schedule, displaying 126
example 126
job, deleting 122
scheduler 122

## L

licensing 166
EEM 166
limitations 195
ERSPAN 195
linkDown notifications 141
linkUp notifications 141
log file size, defining 119
scheduler 119
log file, clearing 124
scheduler 124
log files 118
scheduler 118
logging 77, 82
ACL log match level 82
facility messages 77
module messages 77
logging cache 79
configuring 79

## M

message encryption 134
SNMP 134
mgmt0 interface 80
ACL logging 80
module messages logging 77
configuring 77
MPLS aging 238
MPLS header stripping 231
MPLS overview 231
MPLS stripping feature, enable 236
Multiprotocol Label Switching Overview 231

## N

network taps 229
no isolate 244
no shutdown 244
no system interface shutdown 244
no system mode maintenance 255
no system mode maintenance dont-generate-profile 255
no system mode maintenance on-reload reset-reason 253
notification receivers 135
SNMP 135
NTO on an interface, Enabling and disabling 56

ntp 53–54
information 53
virtualization 54
ntp authenticate 60
ntp authentication-key 60
NTP Broadcast Server, Configuring 64
NTP multicast client, Configuring 66
NTP multicast server, Configuring 65
ntp trusted-key 60

## O

overview 163
embedded event manager 163

## P

password requirements 285
periodic inventory notifications, configuring 105
smart call home 105
policies 164
EEM 164
prerequisites 55, 166, 195, 220
EEM 166
ERSPAN 195
NTP 55
sFlow 220
PTP 41–44, 46
configuring globally 44
default settings 43
device types 41
guidelines and limitations 43
interface, configuring 46
overview 41
process 42
python instance 245, 247

## R

RBAC 283–285, 287, 289–293
feature groups, creating 290
rules 284
user account restrictions 285
user accounts, configuring 287
user role interface policies, changing 291
user role VLAN policies, changing 292
user role VSAN policies, changing 293
user roles 283
user roles and rules, configuring 289
verifying 293
registering 98
smart call home 98
related documents 213
ERSPAN 213
releasing 68
CSF session lock 68
remote user authentication 118
scheduler 118
remote user authentication, configuring 120–121
scheduler 120–121
requirements 285
user passwords 285
roles 283
authentication 283
rollback 113, 116
checkpoint copy 113
creating a checkpoint copy 113
default settings 116
deleting a checkpoint file 113
description 113
example configuration 113
guidelines 113
high availability 113
implementing a rollback 113
limitations 113
reverting to checkpoint file 113
verifying configuration 116
rules 284
RBAC 284
run bash 266
running config, displaying 27
switch profiles 27
runtime diagnostics 155
information 155

## S

sampling data source 226
sFlow 226
sampling rate 221
sFlow 221
SAN admin user, configuring 288
RBAC 288
scheduler 117–122, 124–126
configuration, verifying 125
default settings 118
disabling 124
enabling 119
guidelines and limitations 118
information about 117
job, deleting 122
log file size, defining 119
log file, clearing 124
log files 118
remote user authentication 118
remote user authentication, configuring 120–121
standards 126
timetable, defining 122

scheduler job, creating 125
example 125
scheduler job, scheduling 125
example 125
scheduler jobs, displaying results 126
example 126
serial IDs 93
description 93
server IDs 93
description 93
session manager 113, 115–116
committing a session 115
configuring an ACL session (example) 115
description 113
discarding a session 115
guidelines 113
limitations 113
saving a session 115
verifying configuration 116
verifying the session 115
sFlow 219–228
agent address 225
analyzer address 224
analyzer port 224
configuration example 227
counter poll interval 222
datagram size 223
default settings 220
feature history 228
guidelines 220
prerequisites 220
sampling data source 226
sampling rate 221
show commands 227
show commands 227
sFlow 227
show install packages 266
show interface brief 257
show maintenance on-reload reset-reasons 257
show maintenance profile 257
show maintenance profile maintenance-mode 246, 257
show maintenance profile normal-mode 248, 257
show maintenance timeout 257
show ntp authentication-keys 60
show ntp authentication-status 61
show ntp trusted-keys 60
show ptp counters 52
show running-config mmode 257
show snapshots 249, 257
show snapshots compare 249, 257
show snapshots dump 257
show snapshots sections 257
show startup-config mmode 257
show system mode 254–255, 257
shutdown 244
sleep instance 245
smart call home 89–91, 97–101, 103–108
adding show commands, alert groups 103
alert groups 91
associating alert groups 103
contact information, configuring 99
default settings 98
description 89
destination profile, creating 100
destination profile, modifying 101
destination profiles 90
duplicate message throttling, disabling 106–107
e-mail details, configuring 104
guidelines and limitations 97
message format options 90
periodic inventory notifications 105
prerequisites 97
registering 98
testing the configuration 107
verifying 108
smart call home messages 90, 92
configuring levels 92
format options 90
SMUs 259–261, 263–265, 267
activating packages 263
adding packages 263
committing the active package set 264
deactivating packages 265
described 259
guidelines 261
limitations 261
package management 260
preparing for package installation 261
prerequisites 260
removing packages 265
snapshot create 249
snapshot delete 249
SNMP 127–131, 133–135, 138, 144
access groups 131
configuring users 133
default settings 131
disabling 144
filtering requests 134
functional overview 127
group-based access 131
guidelines and limitations 131
inband access 138
message encryption 134
notification receivers 135
security model 129
trap notifications 128
user synchronization with CLI 130
user-based security 129
SNMP 129
version 3 security features 128
SNMP (Simple Network Management Protocol) 128
versions 128

SNMP notification receivers 136
configuring with VRFs 136
SNMP notifications 137
filtering based on a VRF 137
SNMPv3 128, 134
assigning multiple roles 134
security features 128
soft error recovery 160
software 183, 195
downgrading 183, 195
loss of ERSPAN configurations 195
loss of SPAN configurations 183
source IDs 93
call home event format 93
source ports, characteristics 182
SPAN 182
source ports, configuring 186
SPAN 186
source-interface, configuring 81
syslog 81
SPAN 181–190
activating sessions 188
characteristics, source ports 182
configuration examples 190
configuration loss when downgrading software 183
creating, deleting sessions 184
description, configuring 187
destination ports, characteristics 182
destinations 182
displaying information 189
egress sources 181
Ethernet destination port, configuring 185
guidelines and limitations 183
ingress sources 181
source port channels, configuring 187
source ports, configuring 186
sources for monitoring 181
VLANs, configuring 187
SPAN sources 181
egress 181
ingress 181
standards 126
scheduler 126
switch profile buffer, displaying 21, 29
switch profiles 10, 21–22, 27–29
buffer, displaying 21, 29
configuration sync after reboot 22
example, local and peer sync 27, 29
guidelines and limitations 10
running config, displaying 27
verify and commit, displaying 28
Switched Port Analyzer 181
syslog 81–82, 176
ACL log match level 82
configuring 82
EEM 176
syslog *(continued)*
source-interface, configuring 81
system interface shutdown 244
system message logging 71–72
guidelines and limitations 72
information about 71
system message logging settings 72
defaults 72
system mode maintenance dont-generate-profile 253
system mode maintenance on-reload reset-reason 253
system mode maintenance shutdown 253
system mode maintenance timeout 253
system policies, overriding 175
EEM 175

## T

Tap aggregation overview 230
Tap Aggregation policy, configuring 233
Tap Aggregation, enabling 232
testing the configuration 107
smart call home 107
timetable, defining 122
scheduler 122
trap notifications 128

## U

user account restrictions 285
RBAC 285
user accounts 285–286, 293
guidelines and limitations 286
passwords 285
verifying 293
user policies, defining 168
EEM 168
user role interface policies, changing 291
RBAC 291
user role VLAN policies, changing 292
RBAC 292
user role VSAN policies, changing 293
user roles 283
RBAC 283
user roles and rules, creating 289
RBAC 289
users 283
description 283

## V

verifying 69, 108, 293
NTP configuration 69
RBAC 293
smart call home 108
user accounts 293

verifying MPLS configuration 239
Verifying Tap Aggregation configuration 235
virtualization 54
    ntp 54
VRFs 136–137
    configuring SNMP notification receivers with 136
    filtering SNMP notifications 137
VSH script 174
    defining EEM policies 174
VSH script policies 166, 174
    EEM 166
    registering and activating 174