

Cisco Nexus 3548 Switch NX-OS System Management Configuration Guide, Release 10.6(x)

First Published: 2025-08-13

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

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

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

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

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

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

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

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

© 2025 Cisco Systems, Inc. All rights reserved.



CONTENTS

Trademarks ?

Preface xv

Audience xv

Document Conventions xv

Related Documentation for Cisco Nexus 3500 Series Switches xvi

Documentation Feedback xvi

Communications, Services, and Additional Information xvi

CHAPTER 1 **New and Changed Information** 1

New and Changed Information 1

CHAPTER 2 **Overview** 3

System Management Features 3

Licensing Requirements 5

Supported Platforms 5

CHAPTER 3 **Two-stage Configuration Commit** 7

About Two-stage Configuration Commit 7

Guidelines and Limitations 8

Configuring in Two-Stage Configuration Commit Mode 8

Aborting the Two-Stage Configuration Commit Mode 12

Displaying Commit IDs 12

Rollback Capability 13

Viewing Current Session Configurations 13

CHAPTER 4 **Configuring PTP** 15

Information About PTP 15
PTP Time Distribution Hold 16
PTP Device Types 16
Clock Modes 17
PTP Process 17
High Availability for PTP 18
Guidelines and Limitations for PTP 18
Default Settings for PTP 19
Configuring PTP 20
Configuring PTP Globally 20
Configuring PTP on an Interface 22
PTP Mixed Mode 24
Configuring Multiple PTP Domains 25
Configuring PTP Grandmaster Clock 27
Configuring PTP Cost Interface 29
Configuring clock Identity 30
Configuring a PTP Interface to Stay in a Master State 30
Timestamp Tagging 32
Configure Timestamp Tagging 32
Configure TTAG Marker Packets and Time Interval 33
Verifying the PTP Configuration 35

CHAPTER 5 **Configuring NTP** 37

Information About NTP 37
NTP as a Time Server 38
Distributing NTP Using CFS 38
Clock Manager 38
Virtualization Support 38
Guidelines and Limitations for NTP 38
Default Settings 40
Configuring NTP 40
Configuring NTP Server and Peer 40

Configuring NTP Authentication 42
Configuring NTP Access Restrictions 43
Configuring the NTP Source IP Address 45
Configuring the NTP Source Interface 45
Configuring NTP Logging 46
Enabling CFS Distribution for NTP 47
Commiting NTP Configuration Changes 48
Discarding NTP Configuration Changes 48
Releasing the CFS Session Lock 49
Verifying the NTP Configuration 49
Configuration Examples for NTP 51
Related Documents for NTP 52
Feature History for NTP 52

CHAPTER 6 **Configuring System Message Logging** 53
Information About System Message Logging 53
Syslog Servers 54
Guidelines and Limitations for System Message Logging 54
Default Settings for System Message Logging 55
Configuring System Message Logging 55
Configuring System Message Logging to Terminal Sessions 55
Configuring System Message Logging to a File 58
Configuring Module and Facility Messages Logging 59
Configuring Logging Timestamps 61
Configuring Logging Syslogs Compliant to RFC 5424 62
Configuring Syslog Servers 63
Configuring syslog on a UNIX or Linux System 64
Configuring syslog Server Configuration Distribution 66
Displaying and Clearing Log Files 67
Configuring DOM Logging 68
Enabling DOM Logging 68
Disabling DOM Logging 69
Verifying the DOM Logging Configuration 69
Verifying the System Message Logging Configuration 69

Repeated System Logging Messages 70

CHAPTER 7 **Configuring Smart Call Home 73**

Information About Smart Call Home 73

Smart Call Home Overview 74

Smart Call Home Destination Profiles 74

Smart Call Home Alert Groups 75

Smart Call Home Message Levels 76

Call Home Message Formats 77

Guidelines and Limitations for Smart Call Home 81

Prerequisites for Smart Call Home 81

Default Call Home Settings 81

Configuring Smart Call Home 82

Registering for Smart Call Home 82

Configuring Contact Information 83

Creating a Destination Profile 85

Modifying a Destination Profile 86

Associating an Alert Group with a Destination Profile 87

Adding Show Commands to an Alert Group 88

Configuring E-Mail Server Details 89

Configuring Periodic Inventory Notifications 91

Disabling Duplicate Message Throttling 92

Enabling or Disabling Smart Call Home 92

Testing the Smart Call Home Configuration 93

Verifying the Smart Call Home Configuration 94

Sample Syslog Alert Notification in Full-Text Format 95

Sample Syslog Alert Notification in XML Format 95

CHAPTER 8 **Configuring Session Manager 99**

Information About Session Manager 99

Guidelines and Limitations for Session Manager 99

Configuring Session Manager 100

Creating a Session 100

Configuring ACLs in a Session 100

Verifying a Session 101
Committing a Session 101
Saving a Session 101
Discarding a Session 101
Configuration Example for Session Manager 102
Verifying the Session Manager Configuration 102

CHAPTER 9 Configuring the Scheduler 103
Information About the Scheduler 103
Remote User Authentication 104
Scheduler Log Files 104
Guidelines and Limitations for the Scheduler 104
Default Settings for the Scheduler 104
Configuring the Scheduler 105
Enabling the Scheduler 105
Defining the Scheduler Log File Size 105
Configuring Remote User Authentication 106
Defining a Job 107
Deleting a Job 108
Defining a Timetable 109
Clearing the Scheduler Log File 111
Disabling the Scheduler 112
Verifying the Scheduler Configuration 112
Configuration Examples for the Scheduler 113
Creating a Scheduler Job 113
Scheduling a Scheduler Job 113
Displaying the Job Schedule 113
Displaying the Results of Running Scheduler Jobs 113
Standards for the Scheduler 114

CHAPTER 10 Configuring SNMP 115
Information About SNMP 115
SNMP Functional Overview 115
SNMP Notifications 116

SNMPv3 116
Security Models and Levels for SNMPv1, v2, and v3 116
User-Based Security Model 117
CLI and SNMP User Synchronization 118
Group-Based SNMP Access 119
Guidelines and Limitations for SNMP 119
Default SNMP Settings 119
Configuring SNMP 120
Configuring SNMP Users 120
Enforcing SNMP Message Encryption 121
Assigning SNMPv3 Users to Multiple Roles 121
Creating SNMP Communities 122
Filtering SNMP Requests 122
Configuring SNMP Notification Receivers 123
Configuring SNMP Notification Receivers with VRFs 124
Filtering SNMP Notifications Based on a VRF 124
Configuring SNMP for Inband Access 125
Enabling SNMP Notifications 127
Configuring Link Notifications 128
Disabling Link Notifications on an Interface 129
Enabling One-Time Authentication for SNMP over TCP 130
Assigning SNMP Switch Contact and Location Information 130
Configuring the Context to Network Entity Mapping 131
Disabling SNMP 131
Verifying the SNMP Configuration 132
Additional References 132

CHAPTER 11 **Configuring RMON** 133
Information About RMON 133
RMON Alarms 133
RMON Events 134
Configuration Guidelines and Limitations for RMON 134
Configuring RMON 134
Configuring RMON Alarms 134

Configuring RMON Events 136
Verifying the RMON Configuration 137
Default RMON Settings 137

CHAPTER 12 Configuring Online Diagnostics 139
Information About Online Diagnostics 139
Bootup Diagnostics 139
Health Monitoring Diagnostics 140
Expansion Module Diagnostics 141
Guidelines and Limitations for Online Diagnostics 141
Configuring Online Diagnostics 141
Verifying the Online Diagnostics Configuration 142
Default Settings for Online Diagnostics 143

CHAPTER 13 Configuring Embedded Event Manager 145
About Embedded Event Manager 145
Embedded Event Manager Policies 146
Event Statements 146
Action Statements 147
VSH Script Policies 148
Prerequisites for Embedded Event Manager 148
Guidelines and Limitations for Embedded Event Manager 148
Default Settings for Embedded Event Manager 149
Defining an Environment Variable 149
Defining a User Policy Using the CLI 150
Configuring Event Statements 152
Configuring Action Statements 154
Defining a Policy Using a VSH Script 156
Registering and Activating a VSH Script Policy 157
Overriding a System Policy 158
Configuring Syslog as an EEM Publisher 159
Event Log Auto-Collection and Backup 160
Extended Log File Retention 161
Enabling Extended Log File Retention For All Services 161

Disabling Extended Log File Retention For All Services 162
Enabling Extended Log File Retention For a Single Service 162
Displaying Extended Log Files 164
Disabling Extended Log File Retention For a Single Service 164
Trigger-Based Event Log Auto-Collection 166
Enabling Trigger-Based Log File Auto-Collection 166
Auto-Collection YAML File 166
Limiting the Amount of Auto-Collections Per Component 169
Auto-Collection Log Files 169
Verifying Trigger-Based Log Collection 173
Checking Trigger-Based Log File Generation 173
Local Log File Storage 173
Generating a Local Copy of Recent Log Files 174
External Log File Storage 176

CHAPTER 14 **Configuring SPAN** 179
Information About SPAN 179
Guidelines and Limitations for SPAN 179
SPAN Sources 180
Characteristics of Source Ports 180
SPAN Destinations 180
Characteristics of Destination Ports 181
SPAN and ERSPAN Filtering 181
Guidelines and Limitations for SPAN and ERSPAN Filtering 181
SPAN and ERSPAN Control-packet Filtering 182
SPAN and ERSPAN Sampling 183
Guidelines and Limitations for SPAN and ERSPAN Sampling 183
SPAN and ERSPAN Truncation 183
Guidelines and Limitations for SPAN and ERSPAN Truncation 183
Creating or Deleting a SPAN Session 184
Configuring an Ethernet Destination Port 184
Configuring Source Ports 186
Configuring Source Port Channels or VLANs 186
Configuring the Description of a SPAN Session 187

Activating a SPAN Session 188
Suspending a SPAN Session 188
Configuring a SPAN Filter 189
Configuring SPAN Sampling 190
Configuring SPAN Truncation 192
Displaying SPAN Information 193

CHAPTER 15 Configuring Warp SPAN 195
Information About Warp SPAN 195
Guidelines and Limitations for Warp Span 196
Configuring Warp SPAN 197
Verifying Warp SPAN Mode Configuration 198
Feature History for Warp SPAN 199

CHAPTER 16 Configuring ERSPAN 201
Information About ERSPAN 201
ERSPAN Types 201
ERSPAN Sources 201
ERSPAN Destinations 202
ERSPAN Sessions 202
Multiple ERSPAN Sessions 203
ERSPAN Marker Packet 203
High Availability 203
Prerequisites for ERSPAN 203
Guidelines and Limitations for ERSPAN 204
Default Settings for ERSPAN 205
Configuring ERSPAN 206
Configuring an ERSPAN Source Session 206
Configuring an ERSPAN Destination Session 209
Shutting Down or Activating an ERSPAN Session 211
Configuring ERSPAN Filtering 213
Configuring ERSPAN Sampling 215
Configuring ERSPAN Truncation 217
Configuring an ERSPAN Marker Packet 218

Verifying the ERSPAN Configuration 219
Configuration Examples for ERSPAN 219
Configuration Example for an ERSPAN Source Session 219
Configuration Example for an ERSPAN Destination Session 220
Additional References 220
Related Documents 220

CHAPTER 17 Configuring DNS 221
Information About DNS Client 221
Name Servers 221
DNS Operation 221
High Availability 222
Prerequisites for DNS Clients 222
Default Settings for DNS Clients 222
Configuring DNS Clients 222

CHAPTER 18 Configuring Traffic Forwarding Modes 225
Information About Warp Mode 225
Guidelines and Limitations for Warp Mode 225
Enabling and Disabling Warp Mode 226
Verifying Warp Mode Status 227
Feature History for Warp Mode 227

CHAPTER 19 Configuring Active Buffer Monitoring 229
Information About Active Buffer Monitoring 229
Active Buffer Monitoring Overview 229
Buffer Histogram Data Access and Collection 230
Configuring Active Buffer Monitoring 230
Displaying Buffer Histogram Data 231

CHAPTER 20 Performing Software Maintenance Upgrades (SMUs) 237
About SMUs 237
Package Management 238
Prerequisites for SMUs 238

Guidelines and Limitations for SMUs 239
Performing a Software Maintenance Upgrade for Cisco NX-OS 239
Preparing for Package Installation 239
Copying the Package File to a Local Storage Device or Network Server 241
Adding and Activating Packages 241
Committing the Active Package Set 243
Deactivating and Removing Packages 243
Displaying Installation Log Information 244

CHAPTER 21 Configuration Replace and Commit-timeout 247
Overview 247
Guidelines and Limitations 256
Perform Configuration Replace 260

CHAPTER 22 Configuring Rollback 263
Information About Rollbacks 263
Guidelines and Limitations for Rollbacks 263
Creating a Checkpoint 264
Implementing a Rollback 265
Verifying the Rollback Configuration 266

CHAPTER 23 Candidate Configuration 267
Overview 267
Guidelines and Limitations 269
Perform Integrity Check for Candidate Config 274

CHAPTER 24 Configuring User Accounts and RBAC 277
Information About User Accounts and RBAC 277
User Roles 277
Rules 278
User Role Policies 278
User Account Configuration Restrictions 279
User Password Requirements 279
Guidelines and Limitations for User Accounts 280

Configuring User Accounts 281
Configuring RBAC 282
Creating User Roles and Rules 282
Creating Feature Groups 284
Changing User Role Interface Policies 284
Changing User Role VLAN Policies 285
Verifying the User Accounts and RBAC Configuration 286
Configuring User Accounts Default Settings for the User Accounts and RBAC 287

CHAPTER 25 Configuring Secure Erase 289
Information about Secure Erase 289
Prerequisites for Performing Secure Erase 289
Guidelines and Limitations for Secure Erase 290
Configuring Secure Erase 290

Preface

This preface includes the following sections:

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.
`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 3500 Series Switches

The entire Cisco Nexus 3500 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

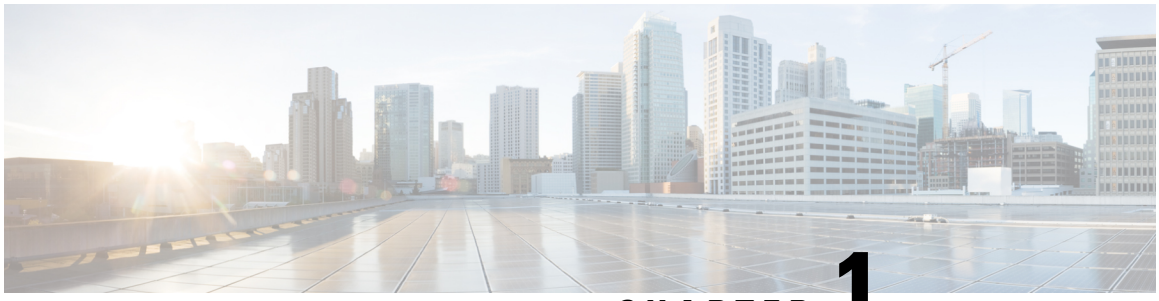
To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus3k-docfeedback@cisco.com. We appreciate your feedback.

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.



CHAPTER 1

New and Changed Information

- New and Changed Information, on page 1

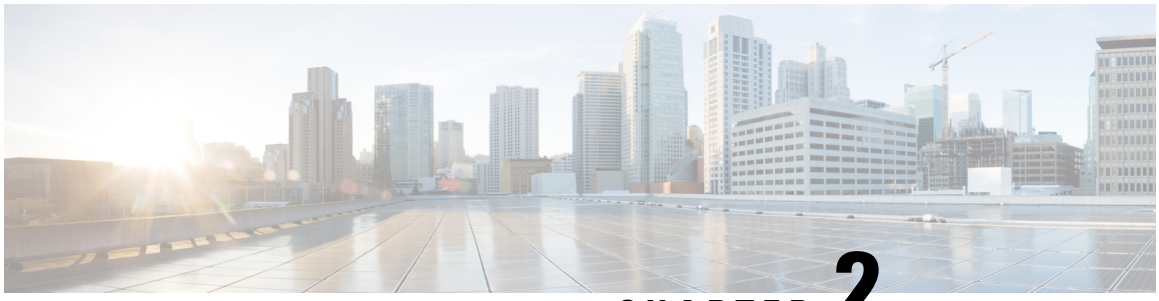
New and Changed Information

This table summarizes the new and changed features for the *Cisco Nexus 3548 Switch NX-OS System Management Configuration Guide, Release 10.6(x)* and where they are documented.

Table 1: New and Changed Features

Feature	Description	Changed in Release	Where Documented
NA	No new features added to this release	10.6(1)F	NA

New and Changed Information



CHAPTER 2

Overview

This chapter contains the following sections:

- System Management Features, on page 3
- Licensing Requirements, on page 5
- Supported Platforms, on page 5

System Management Features

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

Feature	Description
Active Buffer Monitoring	The Active Buffer Monitoring feature provides detailed buffer occupancy data to help you detect network congestion, review past events to understand when and how network congestion is affecting network operations, understand historical trending, and identify patterns of application traffic flow.
Warp Mode	In warp mode, the access path is shortened by consolidating the forwarding table into single table, resulting in faster processing of frames and packets. In warp mode, latency is reduced by up to 20 percent.
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.

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

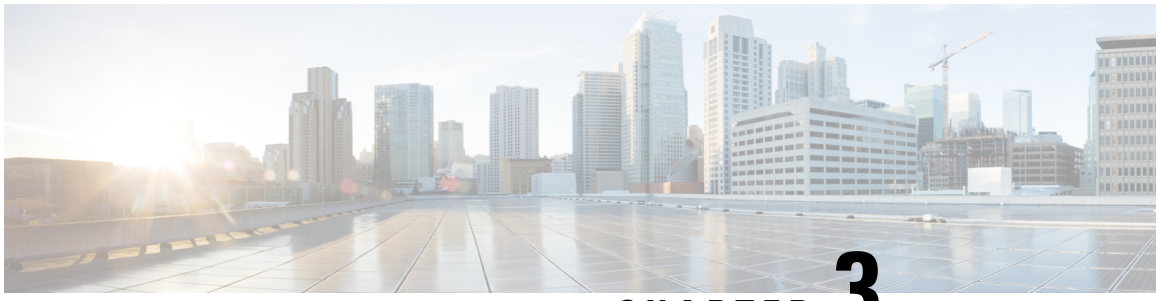
Feature	Description
RMON	RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. Cisco NX-OS supports RMON alarms, events, and logs to monitor Cisco NX-OS devices.
SPAN	The Switched 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.
GIR	Graceful Insertion and Removal (GIR) Maintenance mode is not supported on N3500 Platform.

Licensing Requirements

See the Cisco NX-OS Licensing Guide and Cisco NX-OS Licensing Options Guide for Cisco NX-OS licensing recommendations and instructions to obtain and apply licenses.

Supported Platforms

See the Nexus Switch Platform Support Matrix to know from which Cisco NX-OS releases various Cisco Nexus 9000 and 3000 switches support a selected feature.



CHAPTER 3

Two-stage Configuration Commit

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

This chapter includes the following sections:

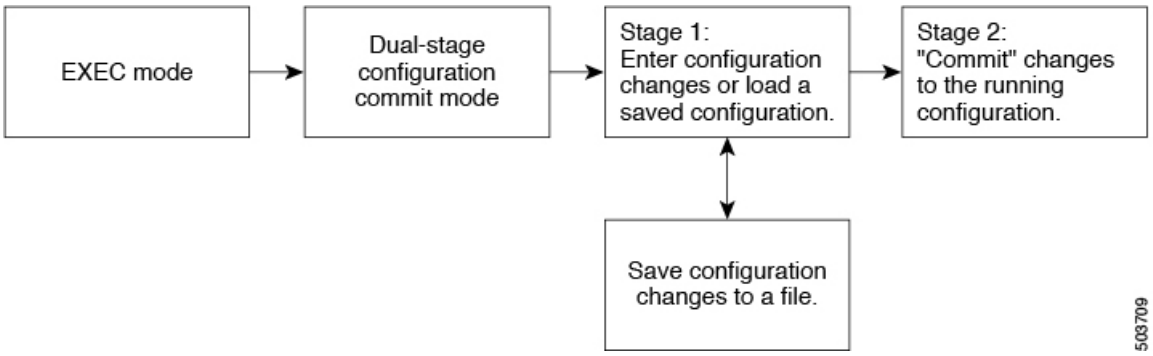
- About Two-stage Configuration Commit, on page 7
- Guidelines and Limitations, on page 8
- Configuring in Two-Stage Configuration Commit Mode, on page 8
- Aborting the Two-Stage Configuration Commit Mode, on page 12
- Displaying Commit IDs, on page 12
- Rollback Capability, on page 13
- Viewing Current Session Configurations, on page 13

About Two-stage Configuration Commit

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

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

Figure 1: Two-Stage Configuration Commit Process



Guidelines and Limitations

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

- This feature is supported only for a CLI interface in a user-interactive session.
- Before you run any feature-related configuration commands, enable the feature using the **feature** command and commit it using the **commit** command.
- Two-stage configuration commit mode doesn't support other modes like maintenance mode, scheduler mode, or virtual mode.
- When you're in the two-stage configuration commit mode, avoid editing configurations in one-stage configuration commit mode from different sessions at the same time.
- Review the configurations using the **show configuration** command before committing the changes.
- If the verification fails, edit and retry the commit.
- If the commit fails, the configuration rolls back to the previous configuration.
- Configurations that you don't commit aren't saved after you reload the switch.
- This feature doesn't support commits with NX-API, EEM, and PPM.
- You can have only one active two-stage configuration commit session at a given time.

Configuring in Two-Stage Configuration Commit Mode

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

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

Procedure

	Command or Action	Purpose

Step 1	**configure dual-stage**
	Example:
	`switch# configure dual-stage`
	`switch(config-dual-stage)#`
		Creates a new target configuration session.
		Note
		The target configuration isn't a copy of the running configuration. It has only the configuration commands entered during the target configuration session.

Step 2	**feature** *feature_name*
	Example:
	`switch(config-dual-stage)# feature bgp`
	`switch(config-dual-stage)#`
		Enables the feature.
		Note
		• You can enable the feature even before entering the two-stage configuration commit mode.
		• You can't combine feature-related commands in a commit if the feature isn't already enabled.

Step 3	**commit** [**confirmed***seconds*]
	Example:
	`switch(config-dual-stage-router)# commit confirmed 30`
	`Verification Succeeded.`
	`Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer.`
	`Please avoid other configuration changes during this time.`
	`Configuration committed by user 'admin' using Commit ID : 1000000001`
	`switch(config-dual-stage)#`
	`switch(config-dual-stage)# commit`
	`Confirming commit for trial session.`
	`switch(config-dual-stage)#`
	Example:
	`switch(config-dual-stage)# hostname example-switch`
	`switch(config-dual-stage)# commit`
	`Verification Succeeded.`
	`Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer.`
	`Please avoid other configuration changes during this time.`
	`Configuration committed by user 'admin' using Commit ID : 1000000002`
	`example-switch(config-dual-stage)#`
		Commits changes to the running configuration.
		• **confirmed**: Commits the changes to the running configuration.
		• *seconds*: Commits the configuration in global configuration mode on a trial basis for a minimum of 30 seconds and a maximum of 65535 seconds.
		Note
		If you enter a trial period, run the commit command to confirm the configuration. If you don't run the commit command, the switch reverts to the previous configuration after the trial period.

Step 4	**Example:**
	`switch(config-dual-stage)# router bgp 64515.46`
	`switch(config-dual-stage-router)#`
		Run any feature-related commands that are supported in this configuration mode.

	Command or Action	Purpose
	`switch(config-dual-stage-router)#  router-id 141.8.139.131` `switch(config-dual-stage-router)#`	

Step 5	**show configuration**
	Example:
	`switch(config-dual-stage-router)# show configuration`
	`! Cached configuration`
	`!`
	`router bgp 64515.46`
	`  router-id 141.8.139.131`
		Displays the target configuration.
		Note
		You can run this command only in the dual-stage configuration mode.

Step 6	**commit** [**confirmed** *seconds*]
	Example:
	`switch(config-dual-stage-router)# commit`
	`Verification Succeeded.`
	`Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer.`
	`Please avoid other configuration changes during this time.`
	`Configuration committed by user 'admin' using Commit ID : 1000000003`
		Commits changes to the running configuration.

Step 7	(Optional) **show configuration commit** [**changes**] *commit-id*
	Example:
	`switch(config-dual-stage-router)# show configuration commit changes 1000000003`
	`*** /bootflash/.dual-stage/1000000003.tmp Fri Mar 19 10:59:00 2021`
	`--- /bootflash/.dual-stage/1000000003  Fri Mar 19 10:59:05 2021`
	`***************`
	`*** 378,383 ****`
	`--- 378,385 ----`
	`  line console`
	`  line vty`
	`  boot nxos bootflash:/nxos64.10.1.1.44.bin`
	`+ router bgp 64515.46`
	`+   router-id 141.8.139.131`
	`  xml server timeout 1200`
	`  no priority-flow-control override-interface mode off`
	Example:
	`switch(config-dual-stage)# show configuration commit 1000000003`
	`feature bgp`
	`router bgp 64515.46`
	`  router-id 141.8.139.131`
	`.`
	`.`
	`.`
		Displays commit-related information.
		Only the last 50 commits or the commit files stored in the reserved disk space are saved. The reserved disk space is 20 MB. All the commit sessions will be removed when you reload the switch. However, the commit IDs aren't removed.
		Use the **show configuration commit changes** *commit-id* command to view only the changes in the current session of the commit you specify.
		Use the **show configuration commit** *commit-id* command to view the complete configurations in the commit you specify.

	Command or Action	Purpose
Step 8	(Optional) **save configuration** *filename* **Example:** `switch(config-dual-stage)# save configuration bootflash:test.cfg`	Saves the target configurations to a separate file without committing them to the running configuration. **Note** • You can load the target configuration files later, modify, or commit. The file will be saved in bootflash. • You can view the configuration file you saved by running the **show configuration file** *filename* command. • Some of the user-specific information will be masked based on the user role.

Step 9	(Optional) **load** *filename*
	Example:
	`switch (config-dual-stage)# show configuration`
	`! Cached configuration`
	`switch (config-dual-stage)# load test.cfg`
	`switch (config-dual-stage-router)# show configuration`
	`! Cached configuration`
	`!`
	`router bgp 1`
	`switch(config-dual-stage-router)#`
		Loads a target configuration that you saved. After loading a file, you can modify it or commit it to the running configuration. To save the changes, use the **save configuration** *filename* command.
		You can load a target configuration that you saved using only the **save configuration** *filename* command.

Step 10	(Optional) **clear configuration**
	Example:
	`switch(config-dual-stage)# show configuration`
	`! Cached configuration`
	`!`
	`router bgp 64515.46`
	`router-id 141.8.139.131`
	`switch (config-dual-stage)# clear configuration`
	`switch (config-dual-stage)# show configuration`
	`! Cached configuration`
	`switch (config-dual-stage)#`
		Clears changes made to the target configuration without terminating the configuration session. It deletes any configuration changes that aren't committed.

Step 11	**end**
	Example:
	`switch(config-dual-stage-if)# end`
	`Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]`
		Exits the global dual stage configuration mode.
		If you end a configuration session without committing the configuration changes, you'll be prompted to save changes, discard changes, or cancel the action:
		• Yes: Commits the configuration changes and exit configuration mode
		• No: Exits the configuration mode without committing the configuration changes
		• Cancel: Remains in configuration mode without committing the configuration changes

	Command or Action	Purpose
		Note • If you choose to exit when a confirm commit timer is running, the same options are displayed. If you still chose to exit, the trial configuration rolls back instantly. • If the default session times out before the timer expires, the trial configuration rolls back before exiting the session. In this case, no warning message appears.

Aborting the Two-Stage Configuration Commit Mode

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

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

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

version 10.1(2) Bios:version
feature bgp


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

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

version 10.1(2) Bios:version
feature bgp

switch#
```

Displaying Commit IDs

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

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

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

Rollback Capability

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

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

:
:

switch#

Each commit ID acts as a checkpoint of a running configuration. You can rollback to any
given commit ID. A new commit ID will be generated after you rollback. If a confirm commit
 session is in progress, you cannot trigger a rollback until it is completed.


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

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

Viewing Current Session Configurations

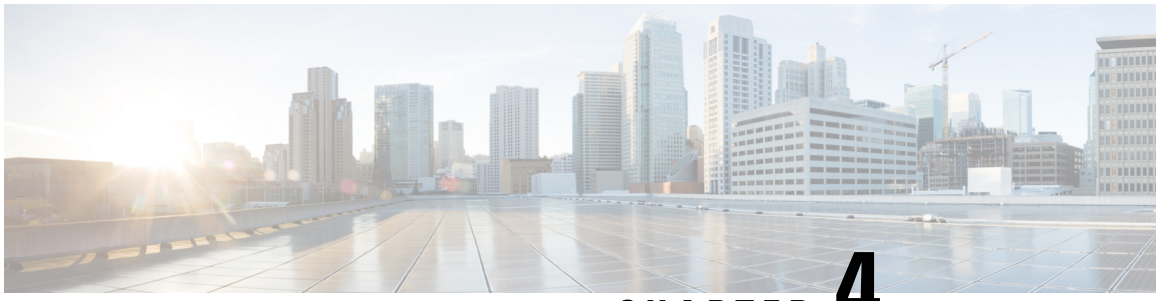
You can view the current session configuration using the **show configuration** command. This command is supported only in the dual-stage mode. The session configuration is cleared if a commit fails.

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

If there is no configuration, the following message appears:

switch(config-dual-stage)# show configuration
! Cached configuration
```

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



CHAPTER 4

Configuring PTP

This chapter contains the following sections:

- Information About PTP, on page 15
- PTP Time Distribution Hold, on page 16
- PTP Device Types, on page 16
- PTP Process, on page 17
- High Availability for PTP, on page 18
- Guidelines and Limitations for PTP, on page 18
- Default Settings for PTP, on page 19
- Configuring PTP, on page 20

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.

Starting from Cisco NXOS Release 6.0(2)A8(3), PTP supports configuring multiple PTP clocking domains, PTP grandmaster capability, PTP cost on interfaces for slave and passive election, and clock identity.

All the switches in a multi-domain environment, belong to one domain. The switches that are the part of boundary clock, must have multi-domain feature enabled on them. Each domain has user configurable parameters such as domain priority, clock class threshold and clock accuracy threshold. The clocks in each domain remain synchronized with the master clock in that domain. If the GPS in a domain fails, the master clock in the domain synchronizes time and data sets associated with the announce messages from the master clock in the domain where the GPS is active. If the master clock from the highest priority domain does not meet the clock quality attributes, a clock in the subsequent domain that match the criteria is selected. The Best Master Clock Algorithm (BMCA) is used to select the master clock if none of the domains has the desired

clock quality attributes. If all the domains have equal priority and the threshold values less than master clock attributes or if the threshold values are greater than the master clock attributes, BMCA is used to select the master clock.

Grandmaster capability feature controls the switch's ability of propagating its clock to other devices that it is connected to. When the switch receives announce messages on an interface, it checks the clock class threshold and clock accuracy threshold values. If the values of these parameters are within the predefined limits, then the switch acts as per PTP standards specified in IEEE 1588v2. If the switch does not receive announce messages from external sources or if the parameters of the announce messages received are not within the predefined limits, the port state will be changed to listening mode. On a switch with no slave ports, the state of all the PTP enabled ports is rendered as listening and on a switch with one slave port, the BMCA is used to determine states on all PTP enabled ports. Convergence time prevents timing loops at the PTP level when grandmaster capability is disabled on a switch. If the slave port is not selected on the switch, all the ports on the switch will be in listening state for a minimum interval specified in the convergence time. The convergence time range is from 3 to 2600 seconds and the default value is 30 seconds.

The interface cost applies to each PTP enabled port if the switch has more than one path to grandmaster clock. The port with the least cost value is elected as slave and the rest of the ports will remain as passive ports.

The clock identity is a unique 8-octet array presented in the form of a character array based on the switch MAC address. The clock identity is determined from MAC according to the IEEE1588v2-2008 specifications. The clock ID is a combination of bytes in a VLAN MAC address as defined in IEEE1588v2.

PTP Time Distribution Hold

In a properly synchronized PTP network, when any PTP node goes down and comes up, the PTP clock is synchronized to its primary time source (GM). During this process, the local node has significant correction and it tries to correct its local clock. At that time, the node can send incorrect time to the downstream nodes and cause issues for all downstream nodes. The Time Distribution (TD) hold feature, introduced in Cisco NX-OS Release 10.5(1)F, resolves this issue by ensuring that the node is properly synchronized to its primary source and distributes time to the downstream nodes during boot up.

The TD hold feature holds the time distribution until a Boundary Clock (BC) node locks to the primary time source and settles down to the target correction value. The TD hold enabled node receives all PTP packets, does the normal state change, and synchronizes time, but it does not send any PTP packets out.



Note If all nodes reboot at the same time (with a difference of few seconds), each node will be in active hold time, which sometimes results in no nodes having secondary port. This leads to the BMC taking a long time to find the best clock. Hence, the user needs to take this into account when enabling this feature.

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.

Clock Modes

The IEEE 1588 standard specifies two clock modes for the PTP supporting devices to operate in: one-step and two-step. However, Nexus 3500 supports only two-step mode.

Two-Step Mode:

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

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

- In a Cisco Nexus 3500 only environment, PTP clock correction is expected to be in the 1- to 2-digit range, from 1 to 99 nanoseconds. However, in a mixed environment, PTP clock correction is expected to be up to 3 digits, from 100 to 999 nanoseconds.
- Cisco Nexus 3500 Series switches support mixed non-negotiated mode of operation on master PTP ports. Meaning that when a slave client sends unicast delay request PTP packet, the Cisco Nexus 3500 responds with an unicast delay response packet. And, if the slave client sends multicast delay request PTP packet, the Cisco Nexus 3500 responds with a multicast delay response packet. For mixed non-negotiated mode to work, the source IP address used in the `ptp source <IP address>` configuration on the BC device must also be configured on any physical or logical interface of the BC device. The recommended best practice is to use the loopback interface of the device.
- Cisco Nexus 3500 Series switches support .
- Cisco Nexus 3500 Series switches do not support PTP on 40G interfaces.
- PTP operates only in boundary clock mode. End-to-end transparent clock and peer-to-peer transparent clock modes are not supported.
- N3500 does not support one-step sync packets.
- When running PTP, without NTP, **clock protocol ptp** needs to be configured.
- 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.

- When **ptp acl-redirect** is configured, PTP management packets are software forwarded. The management packets are expected to be within 100 packets/second to avoid CoPP drops.
- PTP-capable ports do not identify PTP packets and do not time-stamp or redirect those packets to CPU for processing unless you enable PTP on those ports. This means that if the PTP is disabled on a port, then the device will be capable of routing any multicast PTP packets, regardless of their type, assuming that there is a multicast state present for this. None of these multicast PTP packets from this port will be redirected to CPU for processing, because the exception used to redirect them to the CPU is programmed on a per-port basis, based on whether the PTP is enabled or not on the respective port.
- 1 pulse 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 –3 to 1.
- All unicast and multicast PTP management messages will be forwarded as per the forwarding rules. All PTP management messages will be treated as regular multicast packets and process these in the same way as the other non-PTP multicast packets are processed by Cisco Nexus 3500 switches.
- You must configure the incoming port as L3/SVI to enable forwarding of the PTP unicast packets.
- We recommend that Cisco Nexus 3500 switches do not participate in unicast negotiation between the unicast master and clients.
- One-step PTP is not supported on Cisco Nexus 3500 series platform switches.
- Beginning with Cisco NX-OS Release 10.5(2)F, NTP and PTP can coexist on Nexus 3548 platform.
- Beginning with Cisco NX-OS Release 10.5(1)F, the PTP Time Distribution (TD) hold feature is introduced. This feature allows for holding the time distribution until a Boundary Clock node locks to the primary time source and settles down to the target correction value.

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
PTP domain	0. PTP multi domain is disabled by default.
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	1 log second

Parameters	Default
PTP announce timeout	3 announce intervals
PTP minimum delay request interval	1 log second
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.

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **feature ptp**
3. [**no**] **ptp source** *ip-address*
4. (Optional) [**no**] **ptp domain** *number*
5. (Optional) [**no**] **ptp priority1** *value*
6. (Optional) [**no**] **ptp priority2** *value*
7. (Optional) [**no**] **ptp acl-redirect**
8. (Optional) **show ptp brief**
9. (Optional) **show ptp clock**
10. (Optional) [**no**] **ptp time distribution-hold** [**correction-threshold** <corr_limit>] [**delay-threshold** <max_delay_time>]
11. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.
Step 2	[**no**] **feature ptp** **Example:** `switch(config) # feature ptp`	Enables or disables PTP on the device. **Note** Enabling PTP on the switch does not enable PTP on each interface.
Step 3	[**no**] **ptp source** *ip-address*	Configures the source IP address for all PTP packets.

	Command or Action	Purpose
	Example: `switch(config) # ptp source 10.2.3.4`	*ip-address*: IPv4 format.
Step 4	(Optional) [**no**] **ptp domain** *number* **Example:** `switch(config) # ptp domain 24`	Configures the domain number to use for this clock. PTP domains allow you to use multiple independent PTP clocking subdomains on a single network. *number*: Range is from 0 to 128.
Step 5	(Optional) [**no**] **ptp priority1** *value* **Example:** `switch(config) # ptp priority1 10`	Configures the priority1 value to use when advertising this clock. This value overrides the default criteria (clock quality, clock class, and so on) for the best master clock selection. Lower values take precedence. *value*: Range is from 0 to 255.
Step 6	(Optional) [**no**] **ptp priority2** *value* **Example:** `switch(config) # ptp priority2 20`	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. *value*: Range is from 0 to 255.

Step 7	(Optional) [**no**] **ptp acl-redirect**
	Example:
	`switch (config)# ptp acl-redirect`
		Configures the system to forward PTP unicast packets in hardware using ACL entries.
		Note
		Make sure that the TCAM SUP region size is greater than 48 or the following error message is display when you attempt to enter the command:
		`switch(config)# ptp acl-redirect`
		`ERROR: PTP acl-redirect supported only if TCAM sup size is greater than 48`
		`2020 May  6 21:27:04 switch %ACLQOS-SLOT1-2-ACLQOS_OOTR: Tcam resource exhausted:`
		`Need to reconfigure SUP region`

Step 8	(Optional) **show ptp brief** **Example:** `switch(config) # show ptp brief`	Displays the PTP status.
Step 9	(Optional) **show ptp clock** **Example:** `switch(config) # show ptp clock`	Displays the properties of the local clock.
Step 10	(Optional) [**no**] **ptp time distribution-hold** [**correction-threshold** <corr_limit>] [**delay-threshold** <max_delay_time>] **Example:**	Enables the PTP time distribution hold feature. **correction-threshold** - Holds the time distribution until the correction settles down to the given specified correction value provided in nanoseconds.

	Command or Action	Purpose
	`switch(config)# ptp time distribution-hold correction-threshold 90000ns delay threshold 4000s`	**delay-threshold** - Sets the maximum time limit in seconds to hold the time-distribution. However, if correction threshold is met before the delay threshold, time distribution resumes. Default correction threshold is 300 nanoseconds, and default delay threshold is 300 seconds for TOR and 900 seconds for modular chassis. The maximum correction threshold is 100000 nanoseconds, and the maximum delay threshold is 5000 seconds.
Step 11	**copy running-config startup-config** **Example:** `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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **interface ethernet** *slot/port*
3. (Optional) switch(config-if) # [**no**] **ptp announce** {**interval** *log seconds* | **timeout** *count*}
4. (Optional) switch(config-if) # [**no**] **ptp delay request minimum interval** *log seconds*
5. (Optional) switch(config-if) # [**no**] **ptp sync interval** *log seconds*
6. (Optional) switch(config-if) # [**no**] **ptp vlan** *vlan-id*
7. (Optional) switch(config-if) # **show ptp brief**
8. (Optional) switch(config-if) # **show ptp port interface** *interface slot/port*
9. (Optional) switch(config-if)# **copy running-config startup-config**

DETAILED STEPS

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	(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. 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 4	(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 master state. The range is from -1 to -6 log seconds. Where, log (-2) = 4 frames per second.
Step 5	(Optional) switch(config-if) # [**no**] **ptp sync interval** *log seconds*	Configures the interval between PTP synchronization messages on an interface. The range for the PTP synchronization interval is from -3 log second to 1 log second
Step 6	(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. The range is from 1 to 4094.

	Command or Action	Purpose
Step 7	(Optional) switch(config-if) # **show ptp brief**	Displays the PTP status.
Step 8	(Optional) switch(config-if) # **show ptp port interface** *interface slot/port*	Displays the status of the PTP port.
Step 9	(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 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 1/1
PTP Port Dataset: Eth1/1
Port identity: clock identity: f4:4e:05:ff:fe:84:7e:7c
Port identity: port number: 0
PTP version: 2
Port state: Slave
VLAN info: 1
Delay request interval(log mean): 0
Announce receipt time out: 3
Peer mean path delay: 0
Announce interval(log mean): 1
Sync interval(log mean): 1
Delay Mechanism: End to End
Cost: 255
Domain: 5
switch(config-if)#
```

PTP Mixed Mode

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

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # [**no**] **feature ptp**
3. switch(config) # [**no**] **ptp source** *ip-address* [**vrf** *vrf*]
4. switch(config) # [**no**] **ptp multi-domain**
5. switch(config) # [**no**] **ptp domain** *value* **priority** *value*
6. switch(config) # [**no**] **ptp domain** *value* **clock-class-threshold** *value*
7. switch(config) # [**no**] **ptp domain** *value* **clock-accuracy-threshold** *value*
8. switch(config) # [**no**] **ptp multi-domain transition-attributes priority1** *value*
9. switch(config) # [**no**] **ptp multi-domain transition-attributes priority2** *value*
10. switch(config-if) # [**no**] **ptp domain** *value*

DETAILED STEPS

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. **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. 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. The range for the domain *value* is from 0 to 127. The default value of the domain is 0 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. The range for the domain *value* is from 0 to 127.

	Command or Action	Purpose
		The range for the clock-class-threshold *value* is from 0 to 255. **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. The range for the domain *value* is from 0 to 127. The range for the clock-accuracy-threshold *value* is from 0 to 255.
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. 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. 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). 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 PTP Grandmaster Clock

You can configure convergence time to prevent timing loops at the PTP level when grandmaster capability is disabled on a switch. Grandmaster capability is enabled on the device by default.

.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # [**no**] **feature ptp**
3. switch(config) # [**no**] **ptp source** *ip-address* [**vrf** *vrf*]
4. switch(config) # **no ptp grandmaster-capable** [*convergence-time*]
5. switch(config) # [**no**] **ptp domain** *value* **clock-class-threshold** *value*
6. switch(config) # [**no**] **ptp domain** *value* **clock-accuracy-threshold** *value*
7. switch(config) # **ptp grandmaster-capable**

DETAILED STEPS

Procedure

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

	Command or Action	Purpose
Step 2	switch(config) # [**no**] **feature ptp**	Enables or disables PTP on the device. **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. The *ip-address* can be in IPv4 format.
Step 4	switch(config) # **no ptp grandmaster-capable** [*convergence-time*]	Disables grandmaster capability on the switch. Prevents the device from acting as a grandmaster when there is no external grandmaster available in any domains. The default convergence time is 30 seconds.
Step 5	switch(config) # [**no**] **ptp domain** *value* **clock-class-threshold** *value*	Specify the values for domain and clock class threshold. *Clock class threshold* defines the threshold value of clock class that the device uses to determine whether the source clock can be considered as a grandmaster clock. The range for the domain *value* is from 0 to 127. The range for the clock-class-threshold *value* is from 0 to 255. **Note** The switch uses this value to determine whether the source clock is traceable. If the clock class value from all the peers is higher than the clock class threshold value, the BMCA may change all the port state to listening.
Step 6	switch(config) # [**no**] **ptp domain** *value* **clock-accuracy-threshold** *value*	Specify the values for domain and clock accuracy threshold The range for the domain *value* is from 0 to 127. The range for the clock-accuracy-threshold *value* is from 0 to 255.
Step 7	switch(config) # **ptp grandmaster-capable**	Enables grandmaster capability on a switch.

Example

The following example displays the PTP clock information:

```
switch(config-if)# show ptp clock
PTP Device Type: Boundary clock
Clock Identity : f4:4e:05:ff:fe:84:7e:7c
Clock Domain: 5
Number of PTP ports: 2
Priority1 : 129
Priority2 : 255
Clock Quality:
Class : 248
Accuracy : 254
```

```
Offset (log variance) : 65535
Offset From Master : 0
Mean Path Delay : 391
Steps removed : 1
Local clock time:Wed Nov 9 10:31:21 2016
switch(config-if)#
```

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.

.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # [**no**] **feature ptp**
3. switch(config) # [**no**] **ptp source** *ip-address* [**vrf** *vrf*]
4. switch(config) # **interface ethernet** *slot/port*
5. switch(config-if) # [**no**] **ptp cost** *value*

DETAILED STEPS

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. **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. The *ip-address* can be in IPv4 format.
Step 4	switch(config) # **interface ethernet** *slot/port*	Specifies the interface on which you are enabling PTP and enters interface configuration mode.
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. 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)#
```

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.

.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # [**no**] **feature ptp**
3. switch(config-if) # **ptp clock-identity** *MAC Address*

DETAILED STEPS

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. **Note** 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 a PTP Interface to Stay in a Master State

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

Before you begin

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

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

SUMMARY STEPS

1. switch # **configure terminal**
2. switch(config) # **interface ethernet** *slot/port*
3. switch(config) # [**no**] **ptp**
4. switch(config-if) # **ptp transmission multicast**
5. switch(config-if) # **ptp role master**

DETAILED STEPS

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) # [**no**] **ptp**	Enables or disables PTP on an interface.
Step 4	switch(config-if) # **ptp transmission multicast**	Configures the PTP transmission method that is used by the interface.
Step 5	switch(config-if) # **ptp role master**	Configures the PTP role of the interface. **master**: The master clock is assigned as the PTP role of the interface.

Example

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

```
switch(config)# show ptp brief

PTP port status
-----------------------------------
Port                  State
----------------  ----------------
Eth1/1              Slave
switch(config)# interface ethernet 1/1
switch(config-if)# ptp multicast master-only
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_GM_CHANGE: Grandmaster clock has changed
 from 60:73:5c:ff:fe:62:a1:41 to 58:97:bd:ff:fe:0d:54:01 for the PTP protocol
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_STATE_CHANGE: Interface Eth1/1 change from
 PTP_BMC_STATE_SLAVE to PTP_BMC_STATE_PRE_MASTER
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_TIMESYNC_LOST: Lost sync with  master clock
2001 Jan  7 07:50:07 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_STATE_CHANGE: Interface Eth1/1 change from
 PTP_BMC_STATE_PRE_MASTER to PTP_BMC_STATE_MASTER
```

Timestamp Tagging

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

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

Configure Timestamp Tagging

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

Note

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

Procedure

Step 1 Enter the global configuration mode using the command **configure terminal**

Example:

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

Step 2 Enter the interface configuration mode for the specified interface using the command **interface***type slot/port*

Example:

```
switch(config)# interface ethernet 2/2
switch(config-if)#
```

Step 3 Configure the timestamp tagging on the Layer 2 or Layer 3 egress interface using the command [**no**] **ttag**

Example:

```
switch(config-if)# ttag
```

This is required on the ingress port for the traffic that needs to be tagged when egressing the switch. This is not required on the egress port.

Configure TTAG Marker Packets and Time Interval

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

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

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

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

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

Step 2 Configure the seconds that a switch will take to send a ttag-marker packet to the outgoing ports using the command **ttag-marker-interval***seconds*

Example:

```
switch(config-if)# ttag-marker-interval 90
```

This is a global setting to the switch. By default, it sends a ttag-marker packet every 60 seconds. The range for seconds is from 1 to 25200.

Step 3 Enter the interface configuration mode for the specified interface using the command **interface***type slot/port*

Example:

```
switch(config)# interface ethernet 2/2
switch(config-if)#
```

Step 4 Send the ttag-marker packets to the outgoing port using the command [**no**] **ttag-marker enable**

Example:

```
switch(config-if)# ttag-marker enable
```

Step 5 Remove TTAG from egress packets on the interface using the command **ttag-strip**

Example:

```
switch(config-if)# ttag-strip
```

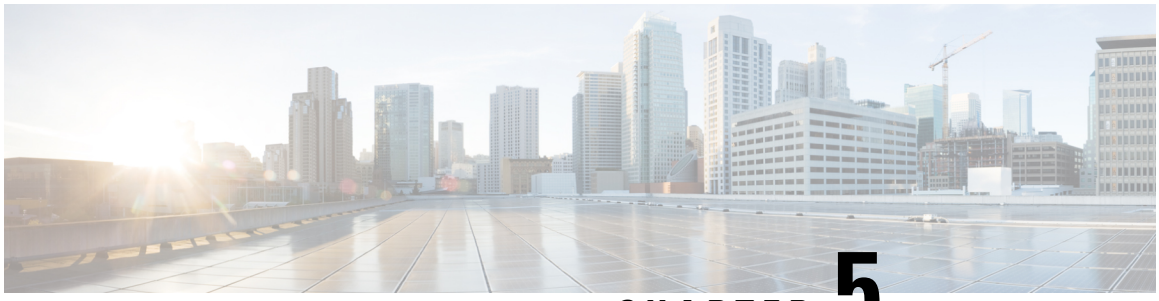
Verifying the PTP Configuration

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.
show ptp port interface ethernet *slot/port*	Displays the status of the PTP port on the switch.
show ptp domain data	Displays multiple domain data, domain priority, clock threshold and information about grandmaster capabilities.
show ptp interface domain	Displays information about the interface to domain association.
show ptp cost	Displays PTP port to cost association.

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



CHAPTER 5

Configuring NTP

This chapter contains the following sections:

- Information About NTP, on page 37
- NTP as a Time Server, on page 38
- Distributing NTP Using CFS, on page 38
- Clock Manager, on page 38
- Virtualization Support, on page 38
- Guidelines and Limitations for NTP, on page 38
- Default Settings, on page 40
- Configuring NTP, on page 40
- Related Documents for NTP, on page 52
- Feature History for NTP, on page 52

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 a Time Server

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

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.

The clock manager allows you to specify the protocol to control the various clocks in the system. Once you specify the protocol, the system clock starts updating.

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.

Guidelines and Limitations for NTP

NTP has the following configuration guidelines and limitations:

- To configure NTP, you must have connectivity to at least one server that is running NTP.
- Configure **clock protocol ntp** to synchronize the system clock from an NTP server, or to act as an NTP server providing time to other devices.
- Beginning with Cisco NX-OS Release 10.4(2)F, NTP and PTP can coexist on Nexus 3548 platform.

- 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, then 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.

Best Practices for NTP configuration

Follow these best practices when configuring NTP:

- Configure multiple redundant NTP time sources to avoid reliance on a single server and improve reliability.
- Use a loopback interface as the NTP source for consistency across device reboots and interface changes.
- Enable NTP authentication to secure communications and prevent malicious time alterations.
- Monitor NTP status and logs regularly; implement detection and alerting mechanisms using commands like **show ntp status** and **show ntp associations**.
- Use the **ntp panic-threshold-apply** command to check the configuration sanity checks, which reject large clock changes and prevent network instability.
- Address failure modes related to time-based authentication keychains by configuring overlapping key lifetimes to avoid routing disruptions.
- Regularly monitor and validate NTP synchronization status with monitoring tools and SNMP traps.
- Ensure operational practices include multiple NTP sources and configuration checks to prevent routing disruptions caused by time inaccuracies.

These practices help maintain accurate time synchronization, improve NTP security, and minimize the risk of network instability caused by incorrect time settings.

Default Settings

Table 4: Default NTP Parameters

Parameters	Default
NTP authentication	disabled
NTP access	enabled
NTP logging	disabled

Configuring NTP

Configuring NTP Server and Peer

You can configure an NTP server and peer.

Before you begin

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

If you plan to use CFS to distribute your NTP configuration to other devices, then you should have already completed the following:

- Enabled CFS distribution.
- Enabled CFS for NTP.

SUMMARY STEPS

1. switch# **configure terminal**
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*]
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*]
4. (Optional) switch(config)# **show ntp peers**
5. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

Procedure

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

	Command or Action	Purpose
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. 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. 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). Use the **prefer** keyword to make this the preferred NTP server for the device. Use the **use-vrf** keyword to configure the NTP server to communicate over the specified VRF. The **vrf-name** argument can be default, management, or any case-sensitive alphanumeric string up to 32 characters. **Note** If you configure a key to be used while communicating with the NTP server, make sure that the key exists as a trusted key on the device.
Step 3	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. 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. 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). Use the **prefer** keyword to make this the preferred NTP server for the device. Use the **use-vrf** keyword to configure the NTP server 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. **Note**

	Command or Action	Purpose
		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.

Example

This example shows how to configure an NTP server and peer:

```
switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp server 192.0.2.10 key 10 use-vrf Red
switch(config)# ntp peer 2001:0db8::4101 prefer use-vrf Red
switch(config)# show ntp peers
-------------------------------------------------
Peer IP Address Serv/Peer
-------------------------------------------------
2001:0db8::4101 Peer (configured)
192.0.2.10 Server (configured)
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

Configuring NTP Authentication

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

Before you begin

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# [**no**] **ntp authentication-key** *number* **md5** *md5-string*
3. (Optional) switch(config)# **show ntp authentication-keys**
4. switch(config)# [**no**]**ntp trusted-key** *number*
5. (Optional) switch(config)# **show ntp trusted-keys**
6. switch(config)# [**no**] **ntp authenticate**
7. (Optional) switch(config)# **show ntp authentication-status**
8. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **ntp authentication-key** *number* **md5** *md5-string*	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.
Step 3	(Optional) switch(config)# **show ntp authentication-keys**	Displays the configured NTP authentication keys.
Step 4	switch(config)# [**no**]**ntp trusted-key** *number*	Specifies one or more keys that a time source must provide in its NTP packets in order for the device to synchronize to it. The range for trusted keys is from 1 to 65535. This command provides protection against accidentally synchronizing the device to a time source that is not trusted.
Step 5	(Optional) switch(config)# **show ntp trusted-keys**	Displays the configured NTP trusted keys.
Step 6	switch(config)# [**no**] **ntp authenticate**	Enables or disables the NTP authentication feature. NTP authentication is disabled by default.
Step 7	(Optional) switch(config)# **show ntp authentication-status**	Displays the status of NTP authentication.
Step 8	(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 synchronize only to time sources that provide authentication key 42 in their NTP packets:

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

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# [**no**] **ntp access-group** {**peer** | **serve** | **serve-only** | **query-only**} *access-list-name*
3. (Optional) switch(config)# **show ntp access-groups**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **ntp access-group** {**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. 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. • The **peer** keyword enables the device to receive time requests and NTP control queries and to synchronize itself to the servers specified in the access list. • The **serve** keyword enables the device to receive time requests and NTP control queries from the servers specified in the access list but not to synchronize itself to the specified servers. • The **serve-only** keyword enables the device to receive only time requests from servers specified in the access list. • The **query-only** keyword enables the device to receive only NTP control queries from the servers specified in the access list.
Step 3	(Optional) switch(config)# **show ntp access-groups**	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# config t
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.

To configure the NTP source IP address, use the following command in global configuration mode:

SUMMARY STEPS

1. switch(config)# [**no**] **ntp source** *ip-address*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch(config)# [**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 NTP to a source IP address:

```
switch(config)# ntp source 192.0.2.1
```

Configuring the NTP Source Interface

You can configure NTP to use a specific interface.

To configure the NTP source interface, use the following command in global configuration mode:

SUMMARY STEPS

1. switch(config)# [**no**] **ntp source-interface** *interface*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch(config)# [**no**] **ntp source-interface** *interface*	Configures the source interface for all NTP packets. Use the **?** keyword to display a list of supported interfaces.

Example

This example shows how to configure NTP to a specific interface:

```
switch(config)# ntp source-interface
ethernet 2/1
```

Configuring NTP Logging

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# [**no**] **ntp logging**
3. (Optional) switch(config)# **show ntp logging-status**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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

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

```
switch# config t
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# [**no**] **ntp distribute**
3. (Optional) switch(config)# **show ntp status**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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 CFS distribution for NTP:

```
switch# config t
Enter configuration commands, one per
line. End with CNTL/Z.
```

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

Commiting 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **ntp commit**

DETAILED STEPS

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.

Example

This example shows how to commit the NTP configuration changes:

```
switch(config)# ntp commit
```

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.

To discard NTP configuration changes, use the following command in global configuration mode:

SUMMARY STEPS

1. switch(config)# **ntp abort**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	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.

Example

This example shows how to discard the NTP configuration changes:

```
switch(config)# ntp abort
```

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.

To release the session lock from any device and discard any pending database changes, use the following command in global configuration mode:

SUMMARY STEPS

1. switch(config)# **clear ntp session**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch(config)# **clear ntp session**	Discards the NTP configuration changes in the pending database and releases the CFS lock.

Example

This example shows how to release the CFS session lock:

```
switch(config)# clear ntp session
```

Verifying the NTP Configuration

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

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

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

SUMMARY STEPS

1. **show ntp access-groups**
2. **show ntp authentication-keys**
3. **show ntp authentication-status**
4. **show ntp logging-status**
5. **show ntp peer-status**
6. **show ntp peers**
7. **show ntp pending**
8. **show ntp pending-diff**
9. **show ntp rts-update**
10. **show ntp session status**
11. **show ntp source**
12. **show ntp source-interface**
13. **show ntp statistics** {**io** | **local** | **memory** | **peer** {**ipaddr** {*ipv4-addr* | *ipv6-addr*} | **name** *peer-name*}}
14. **show ntp status**
15. **show ntp trusted-keys**
16. **show running-config ntp**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**show ntp access-groups**	Displays the NTP access group configuration.
Step 2	**show ntp authentication-keys**	Displays the configured NTP authentication keys.
Step 3	**show ntp authentication-status**	Displays the status of NTP authentication.
Step 4	**show ntp logging-status**	Displays the NTP logging status.
Step 5	**show ntp peer-status**	Displays the status for all NTP servers and peers.
Step 6	**show ntp peers**	Displays all the NTP peers.
Step 7	**show ntp pending**	Displays the temporary CFS database for NTP.
Step 8	**show ntp pending-diff**	Displays the difference between the pending CFS database and the current NTP configuration.
Step 9	**show ntp rts-update**	Displays the RTS update status.
Step 10	**show ntp session status**	Displays the NTP CFS distribution session information.
Step 11	**show ntp source**	Displays the configured NTP source IP address.
Step 12	**show ntp source-interface**	Displays the configured NTP source interface.

	Command or Action	Purpose
Step 13	**show ntp statistics** {**io** | **local** | **memory** | **peer** {**ipaddr** {*ipv4-addr* | *ipv6-addr*} | **name** *peer-name*}}	Displays the NTP statistics.
Step 14	**show ntp status**	Displays the NTP CFS distribution status.
Step 15	**show ntp trusted-keys**	Displays the configured NTP trusted keys.
Step 16	**show running-config ntp**	Displays NTP information.

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 configuration in startup so that it is saved across reboots and restarts:

```
switch# config 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 2001:0db8::4101
switch(config)# show ntp peers
-------------------------------------------------
     Peer IP Address            Serv/Peer
-------------------------------------------------
     2001:db8::4101          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# config 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
```

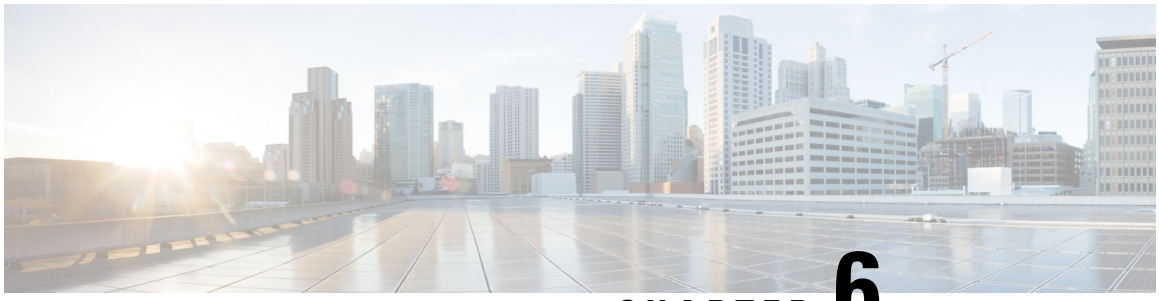
Related Documents for NTP

Related Topic	Document Title
NTP CLI commands	*Cisco Nexus 3548 Switch NX-OS System Management Command Reference Guide*

Feature History for NTP

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Feature Name	Releases	Feature Information
NTP	5.0(3)A1(1)	This feature was introduced.



CHAPTER 6

Configuring System Message Logging

This chapter contains the following sections:

- Information About System Message Logging, on page 53
- Guidelines and Limitations for System Message Logging, on page 54
- Default Settings for System Message Logging, on page 55
- Configuring System Message Logging, on page 55
- Configuring DOM Logging, on page 68
- Verifying the System Message Logging Configuration, on page 69
- Repeated System Logging Messages, on page 70

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 5: 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

Level	Description
5 – notification	Normal but significant condition
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 send 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

System message logging has the following configuration guidelines and limitations:

- System messages are logged to the console and the logfile by default.
- Beginning with Cisco NX-OS Release 10.3(4a)M, the existing **logging rfc-strict 5424** command (optional) that enables the syslog protocol RFC 5424 is enhanced by adding a new keyword (**full**) as follows:

  logging rfc-strict 5424 full

  The addition of this keyword ensures complete compliance with the RFC 5424 standard for Syslog Protocol. However, if the values are not available for the `[APP-NAME] [PROCID] [MSG-ID] [STRUCTRED-DATA]` fields, then the nil value is indicated by a dash (-).
- Beginning with Cisco NX-OS Release 10.5(3), the existing **logging rfc-strict 5424** command (optional) that enables the syslog protocol RFC 5424 is enhanced by adding a new keyword (**utc**) as follows:

  logging rfc-strict 5424 utc

  The addition of this keyword enables the RFC 5424 standard for Syslog Protocol with UTC time format.

  You can also ensure complete compliance with the RFC 5424 standard for Syslog Protocol in UTC time format with the following command: **logging rfc-strict 5424 utc full**.

Default Settings for System Message Logging

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

Table 6: Default System Message Logging Parameters

Parameters	Default
Console logging	Enabled at severity level 2
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.

SUMMARY STEPS

1. switch# **terminal monitor**
2. switch# **configure terminal**
3. switch(config)# **logging console** [*severity-level*]
4. (Optional) switch(config)# **no logging console** [*severity-level*]
5. switch(config)# **logging monitor** [*severity-level*]
6. (Optional) switch(config)# **no logging monitor** [*severity-level*]
7. (Optional) switch# **show logging console**
8. (Optional) switch# **show logging monitor**
9. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging If the severity level is not specified, the default of 2 is used.
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: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging

	Command or Action	Purpose
		If the severity level is not specified, the default of 2 is used. 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **logging logfile** *logfile-name severity-level* [**size** *bytes*]
3. (Optional) switch(config)# **no logging logfile** [*logfile-name severity-level* [**size** *bytes*]]
4. (Optional) switch# **show logging info**
5. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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. Severity levels range from 0 to 7: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging 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.

	Command or Action	Purpose
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
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **logging module** [*severity-level*]
3. switch(config)# **logging level** *facility severity-level*
4. (Optional) switch(config)# **no logging module** [*severity-level*]
5. (Optional) switch(config)# **no logging level** [*facility severity-level*]
6. (Optional) switch# **show logging module**
7. (Optional) switch# **show logging level** [*facility*]
8. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging If the severity level is not specified, the default of 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: • 0 – emergency • 1 – alert • 2 – critical • 3 – error • 4 – warning • 5 – notification • 6 – informational • 7 – debugging To apply the same severity level to all facilities, use the all facility. For defaults, see the **show logging level** command. **Note** If the default severity and current session severity of a component is the same, then the logging level for the component will not be displayed in the running configuration.

	Command or Action	Purpose
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **logging timestamp** {**microseconds** | **milliseconds** | **seconds**}
3. (Optional) switch(config)# **no logging timestamp** {**microseconds** | **milliseconds** | **seconds**}
4. (Optional) switch# **show logging timestamp**
5. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

Procedure

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

	Command or Action	Purpose
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 Logging Syslogs Compliant to RFC 5424

The command can be modified in the following ways :

- [*no*] **logging rfc-strict 5424**
- **show logging rfc-strict 5424**

SUMMARY STEPS

1. `switch(config)#`[*no*] logging rfc-strict 5424
2. `switch(config)#` **logging rfc-strict 5424**
3. `switch(config)#`show logging rfc-strict 5424

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	`switch(config)#`[*no*] logging rfc-strict 5424	(optional) Negate a command or set its defaults
Step 2	`switch(config)#` **logging rfc-strict 5424**	Modify message logging facilities and set RFC to which messages should be compliant.
Step 3	`switch(config)#`show logging rfc-strict 5424	Displays the syslogs which will be compliant to RFC 5424

Configuring Syslog Servers

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

SUMMARY STEPS

1. **configure terminal**
2. **logging server** *host* [*severity-level* [**use-vrf** *vrf-name* [**facility** *facility*]]]
3. (Optional) **no logging server** *host*
4. (Optional) **show logging server**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**logging server** *host* [*severity-level* [**use-vrf** *vrf-name* [**facility** *facility*]]] **Example:** `switch(config)# logging server 172.28.254.254 5 use-vrf default facility local3`	Configures a host to receive syslog messages. • The *host* argument identifies the hostname or the IPv4 or IPv6 address of the syslog server host. • 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 5: System Message Severity Levels , on page 53. • 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. The **show running** command output can display or not display the VRF based on the following configuration scenarios: • 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. • You have configured management VRF. Then this VRF is not displayed in the output as the system identifies it as the default. • You have configured any other VRF. Then this VRF is displayed in the output.

	Command or Action	Purpose
		Note 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. • The facility argument names the syslog facility type. The default outgoing facility is local7. The facilities are listed in the command reference for the Cisco Nexus Series software that you are using. **Note** Debugging is a CLI facility but the debug syslogs are not sent to the server.
Step 3	(Optional) **no logging server** *host* **Example:** `switch(config)# no logging server 172.28.254.254 5`	Removes the logging server for the specified host.
Step 4	(Optional) **show logging server** **Example:** `switch# show logging server`	Displays the syslog server configuration.
Step 5	(Optional) **copy running-config startup-config** **Example:** `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 7: syslog Fields in syslog.conf

Field	Description
Facility	Creator of the message, which can be auth, authpriv, cron, daemon, kern, lpr, mail, mark, news, syslog, user, local0 through local7, or an asterisk (*) for all. These facility designators allow you to control the destination of messages based on their origin. **Note** Check your configuration before using a local facility.
Level	Minimum severity level at which messages are logged, which can be debug, info, notice, warning, err, crit, alert, emerg, or an asterisk (*) for all. You can use none to disable a facility.
Action	Destination for messages, which can be a filename, a hostname preceded by the at sign (@), or a comma-separated list of users or an asterisk (*) for all logged-in users.

SUMMARY STEPS

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:
2. Create the log file by entering these commands at the shell prompt:
3. Make sure that the system message logging daemon reads the new changes by checking myfile.log after entering this command:

DETAILED STEPS

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **logging distribute**
3. switch(config)# **logging commit**
4. switch(config)# **logging abort**
5. (Optional) switch(config)# **no logging distribute**
6. (Optional) switch# **show logging pending**
7. (Optional) switch# **show logging pending-diff**
8. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

Procedure

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

	Command or Action	Purpose
		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.

SUMMARY STEPS

1. switch# **show logging last** *number-lines*
2. switch# **show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]
3. switch# **show logging nvram** [**last** *number-lines*]
4. switch# **clear logging logfile**
5. switch# **clear logging nvram**

DETAILED STEPS

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 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
```

Configuring DOM Logging

Enabling DOM Logging

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **system ethernet dom polling**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **system ethernet dom polling**	Enables transceiver digital optical monitoring periodic polling.

Example

The following example shows how to enable DOM logging.

```
switch# configure terminal
switch(config)# system ethernet dom polling
```

Disabling DOM Logging

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **no system ethernet dom polling**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **no system ethernet dom polling**	Disables transceiver digital optical monitoring periodic polling.

Example

The following example shows how to disable DOM logging.

```
switch# configure terminal
switch(config)# no system ethernet dom polling
```

Verifying the DOM Logging Configuration

Command	Purpose
show system ethernet dom polling status	Displays the transceiver digital optical monitoring periodic polling status.

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.

Command	Purpose
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.
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.

Repeated System Logging Messages

System processes generate logging messages. Depending on the filters used to control which severity levels are generated, a large number of messages can be produced with many of them being repeated.

To make it easier to develop scripts to manage the volume of logging messages, and to eliminate repeated messages from “flooding” the output of the **show logging log** command, the following method of logging repeated messages is used.

In the old method, when the same message was repeated, the default was to state the number of times it reoccurred in the message:

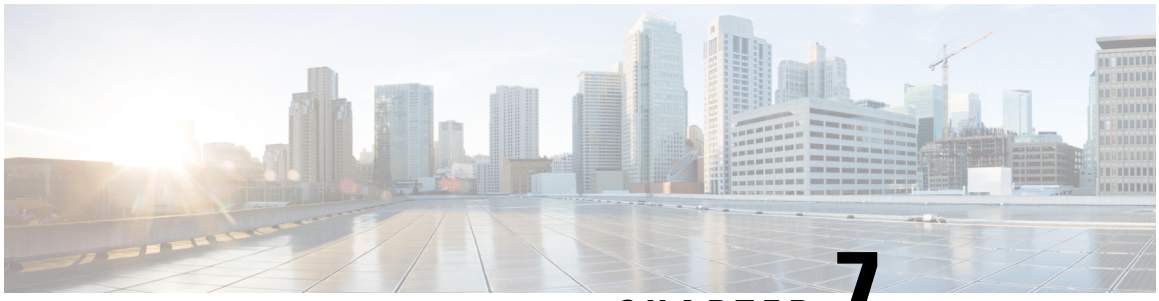
```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5
2019 Mar 11 13:43:15 Cisco-customer last message repeated 242 times
```

The new method simply appends the repeat count to the end of the repeated message:

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5

2019 Mar 11 13:43:15 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
```

```
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5 (message repeated 242 times)
```



CHAPTER 7

Configuring Smart Call Home

This chapter contains the following sections:

- Information About Smart Call Home, on page 73
- Guidelines and Limitations for Smart Call Home, on page 81
- Prerequisites for Smart Call Home, on page 81
- Default Call Home Settings, on page 81
- Configuring Smart Call Home, on page 82
- Verifying the Smart Call Home Configuration, on page 94
- Sample Syslog Alert Notification in Full-Text Format, on page 95
- Sample Syslog Alert Notification in XML Format, on page 95

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 8: 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** **show moduleshow version** **show tech-support platform callhome**
Supervisor hardware	Events related to supervisor modules.	**show diagnostic result module all detail** **show moduleshow version** **show tech-support platform callhome**
Linecard hardware	Events related to standard or intelligent switching modules.	**show diagnostic result module all detail** **show moduleshow version** **show tech-support platform callhome**
Configuration	Periodic events related to configuration.	**show version** **show module** **show running-config all** **show startup-config**
System	Events generated by a failure of a software system that is critical to unit operation.	**show system redundancy status** **show tech-support**
Environmental	Events related to power, fan, and environment-sensing elements such as temperature alarms.	**show environment** **show logging last 1000** **show module show version** **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** **show version** **show license usage** **show inventory** **show sprom all** **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 9: 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 10: 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 11: 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: *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*: • *type* is the product model number from backplane IDPROM. • @ is a separator character. • *Sid* is C, identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. 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. The format is *type@Sid@serial*: • *type* is the product model number from backplane IDPROM. • @ is a separator character. • *Sid* is C, identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. 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 12: 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 13: 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 14: 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.

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 15: 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

Parameters	Default
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

SUMMARY STEPS

1. In a browser, navigate to the Smart Call Home web page:
2. Under **Getting Started**, follow the directions to register Smart Call Home.

DETAILED STEPS

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **snmp-server contact** *sys-contact*
3. switch(config)# **callhome**
4. switch(config-callhome)# **email-contact** *email-address*
5. switch(config-callhome)# **phone-contact** *international-phone-number*
6. switch(config-callhome)# **streetaddress** *address*
7. (Optional) switch(config-callhome)# **contract-id** *contract-number*
8. (Optional) switch(config-callhome)# **customer-id** *customer-number*
9. (Optional) switch(config-callhome)# **site-id** *site-number*
10. (Optional) switch(config-callhome)# **switch-priority** *number*
11. (Optional) switch# **show callhome**
12. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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. The *email-address* can be up to 255 alphanumeric characters in an e-mail address format. **Note** 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. **Note** The phone number cannot contain spaces. Use the plus (+) prefix before the number.

	Command or Action	Purpose
Step 6	switch(config-callhome)# **streetaddress** *address*	Configures the street address for the primary person responsible for the switch. 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. The *contract-number* can be up to 255 alphanumeric characters.
Step 8	(Optional) switch(config-callhome)# **customer-id** *customer-number*	Configures the customer number for this switch from the service agreement. 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. 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. The range is from 0 to 7, with 0 being the highest priority and 7 the lowest. The default is 7. **Note** Switch priority is used by the operations personnel or TAC support personnel to decide which Call Home message should be responded to first. You can prioritize Call Home alerts of the same severity from each switch.
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**
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**}}}
4. (Optional) switch# **show callhome destination-profile** [**profile** *name*]
5. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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** {**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. 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**
3. switch(config-callhome)# **destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **email-addr** *address*
4. **destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **message-level** *number*
5. switch(config-callhome)# **destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **message-size** *number*
6. (Optional) switch# **show callhome destination-profile** [**profile** *name*]
7. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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** {*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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**
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**}
4. (Optional) switch# **show callhome destination-profile** [**profile** *name*]

5. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**
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*
4. (Optional) switch# **show callhome user-def-cmds**
5. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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)# **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. **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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**
3. switch(config-callhome)# **transport email smtp-server** *ip-address* [**port** *number*] [**use-vrf** *vrf-name*]
4. (Optional) switch(config-callhome)# **transport email from** *email-address*
5. (Optional) switch(config-callhome)# **transport email reply-to** *email-address*

6. (Optional) switch# **show callhome transport-email**
7. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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. The *number* range is from1 to 65535. The default port number is 25. 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**
3. switch(config-callhome)# **periodic-inventory notification** [**interval** *days*] [**timeofday** *time*]
4. (Optional) switch# **show callhome**
5. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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. The **interval** *days* range is from 1 to 30 days. The default is 7 days. 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**
3. switch(config-callhome) # **no duplicate-message throttle**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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. 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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**

3. switch(config-callhome) # [**no**] **enable**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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. 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **callhome**
3. switch(config-callhome) # **callhome send diagnostic**
4. switch(config-callhome) # **callhome test**

5. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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.

Command	Purpose
show startup-config callhome	Displays the startup configuration for Smart Call Home.
show tech-support callhome	Displays the technical support output for Smart Call Home.

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:2018-02-08T11:10:44
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:2018 Feb  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

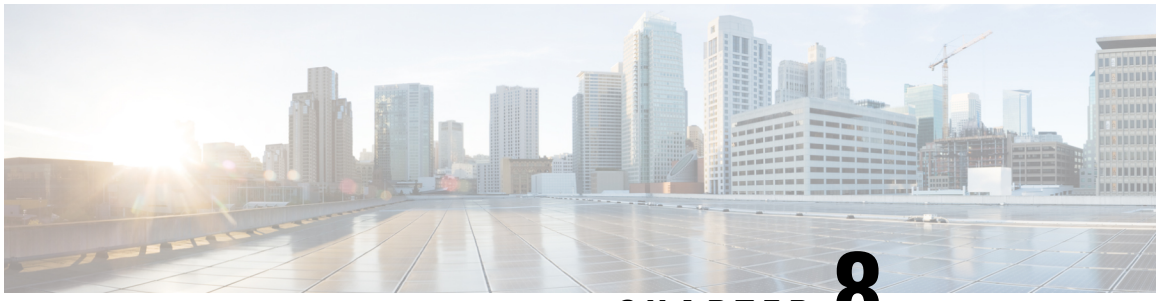
This sample shows the XML format for a syslog port alert-group notification:

```
From: example
Sent: Wednesday, Feb 25, 2018 7:20 AM
To: User (user)
Subject: System Notification From Router - syslog - 2018-02-25 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>2018-02-25 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>2018-02-25 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-Feb-18 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) 1986-2018 by Cisco Systems, Inc.
Compiled Thu 26-Feb-18 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-Feb-18 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-Feb-18 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 8

Configuring Session Manager

This chapter contains the following sections:

- Information About Session Manager, on page 99
- Guidelines and Limitations for Session Manager, on page 99
- Configuring Session Manager, on page 100
- Verifying the Session Manager Configuration, on page 102

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.

SUMMARY STEPS

1. switch# **configure session** *name*
2. (Optional) switch(config-s)# **show configuration session** [*name*]
3. (Optional) switch(config-s)# **save** *location*

DETAILED STEPS

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. 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.

SUMMARY STEPS

1. switch# **configure session** *name*
2. switch(config-s)# **ip access-list** *name*
3. (Optional) switch(config-s-acl)# **permit** *protocol source destination*
4. switch(config-s-acl)# **interface** *interface-type number*
5. switch(config-s-if)# **ip port access-group** *name* **in**
6. (Optional) switch# **show configuration session** [*name*]

DETAILED STEPS

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

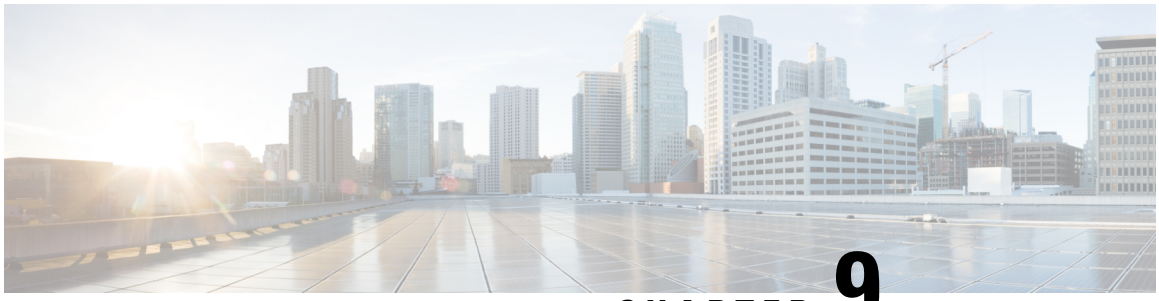
The following example shows how to create a configuration session for ACLs:

```
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 9

Configuring the Scheduler

This chapter contains the following sections:

- Information About the Scheduler, on page 103
- Guidelines and Limitations for the Scheduler, on page 104
- Default Settings for the Scheduler, on page 104
- Configuring the Scheduler, on page 105
- Verifying the Scheduler Configuration, on page 112
- Configuration Examples for the Scheduler, on page 113
- Standards for the Scheduler, on page 114

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 16: Default Command Scheduler Parameters

Parameters	Default
Scheduler state	Disabled
Log file size	16 KB

Configuring the Scheduler

Enabling the Scheduler

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **feature scheduler**
3. (Optional) switch(config) # **show scheduler config**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **scheduler logfile size** *value*
3. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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. The range is from 16 to 1024. The default log file size is 16. **Note** If the size of the job output is greater than the size of the log file, the output is truncated.
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **scheduler aaa-authentication password** [**0** | **7**] *password*
3. switch(config) # **scheduler aaa-authentication username** *name* **password** [**0** | **7**] *password*
4. (Optional) switch(config) # **show running-config** | **include "scheduler aaa-authentication"**
5. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

Procedure

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

	Command or Action	Purpose
Step 2	switch(config) # **scheduler aaa-authentication password** [**0** | **7**] *password*	Configures a password for the user who is currently logged in. To configure a clear text password, enter **0**. 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, you must delete it and create a new one.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **scheduler job name** *name*
3. switch(config-job) # *command1* **;** [*command2* **;***command3* **;** ...
4. (Optional) switch(config-job) # **show scheduler job** [*name*]
5. (Optional) switch(config-job) # **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **scheduler job name** *name*	Creates the job with the specified name and enters the job configuration mode.

	Command or Action	Purpose
		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 (;). Creates the filename using the current timestamp and switch name.
Step 4	(Optional) switch(config-job) # **show scheduler job** [*name*]	Displays the job information. 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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **no scheduler job name** *name*
3. (Optional) switch(config-job) # **show scheduler job** [*name*]
4. (Optional) switch(config-job) # **copy running-config startup-config**

DETAILED STEPS

Procedure

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

	Command or Action	Purpose
Step 2	switch(config) # **no scheduler job name** *name*	Deletes the specified job and all commands defined within it. 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **scheduler schedule name** *name*
3. switch(config-schedule) # **job name** *name*
4. switch(config-schedule) # **time daily** *time*

5. switch(config-schedule) # **time weekly** [[*day-of-week*:] *HH*:] *MM*
6. switch(config-schedule) # **time monthly** [[*day-of-month*:] *HH*:] *MM*
7. switch(config-schedule) # **time start** {**now repeat** *repeat-interval* | *delta-time* [**repeat** *repeat-interval*]}
8. (Optional) switch(config-schedule) # **show scheduler config**
9. (Optional) switch(config-schedule) # **copy running-config startup-config**

DETAILED STEPS

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. 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. 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. 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**). 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. 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. The start-time format is [[[[yyyy:]mmm:]dd:]HH]:MM. • *delta-time*— Specifies the amount of time to wait after the schedule is configured before starting a job. • **now**— Specifies that the job starts two minutes from now. • **repeat** *repeat-interval*— Specifies the frequency at which the job is repeated.

	Command or Action	Purpose
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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **clear scheduler logfile**

DETAILED STEPS

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **no feature scheduler**
3. (Optional) switch(config) # **show scheduler config**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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 17: 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

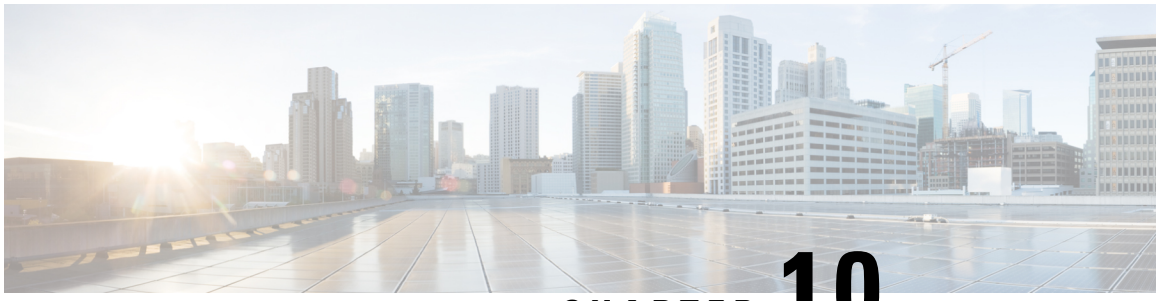
This example shows how to display the results of scheduler jobs that have been executed by the scheduler:

```
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 10

Configuring SNMP

This chapter contains the following sections:

- Information About SNMP, on page 115
- Guidelines and Limitations for SNMP, on page 119
- Default SNMP Settings, on page 119
- Configuring SNMP, on page 120
- Disabling SNMP, on page 131
- Verifying the SNMP Configuration, on page 132
- Additional References, on page 132

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 18: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	authNoPriv	HMAC-MD5, HMAC-SHA, or SHA-256	No	Provides authentication based on the Hash-Based Message Authentication Code (HMAC) Message Digest 5 (MD5) algorithm or the HMAC Secure Hash Algorithm (SHA).
v3	authPriv	HMAC-MD5, HMAC-SHA, or SHA-256	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides Data Encryption Standard (DES) 56-bit encryption in addition to authentication based on the Cipher Block 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
- SHA-256 authentication protocol

Beginning with Cisco NX-OS release 9.3(7), HMAC-SHA-256 authentication protocol is used for SNMPv3.

Note When SHA-256 SNMP users are configured on the switch, ISSD is recommended by **install all** cmd else there will be config loss.

Cisco NX-OS uses Advanced Encryption Standard (AES) as one of the privacy protocols for SNMPv3 message encryption and conforms with RFC 3826.

The **priv** option offers a choice of DES or 128-bit AES encryption for SNMP security encryption. The **priv** option and the **aes-128** token 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

- Commands configured using SNMP SET should be deleted using SNMP SET only. Commands configured using Command Line Interface(CLI) or NX-API should be deleted using CLI or NX-API only.
- Cisco NX-OS supports read-only access to Ethernet MIBs. For more information about supported MIBs, see the following URL:

  ftp://ftp.cisco.com/pub/mibs/supportlists/nexus3000/Nexus3000MIBSupportList.html
- Cisco NX-OS does not support the SNMPv3 noAuthNoPriv security level.
- Cisco Nexus 3548 switches support upto 10000 flash files for *snmpwalk* request.
- Beginning with Cisco NX-OS Release 10.5(2), users can configure AES-256 as the privacy protocol for SNMPv3.
  - Before downgrading to the earlier releases, reconfigure the existing user with encryption AES-256 to AES-128 or remove the user with encryption AES-256.
  - This feature is supported on all N9K platforms.

Default SNMP Settings

Table 19: Default SNMP Parameters

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

Configuring SNMP

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.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server user** *name* [**auth** {**md5** | **sha** | **sha-256**} *passphrase* [**auto**] [**priv [aes-256]** *passphrase*] [**engineID** *id*] [**localizedkey**] [**localizedV2key**]]
3. (Optional) switch# **show snmp user**
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**snmp-server user** *name* [**auth** {**md5** | **sha** | **sha-256**} *passphrase* [**auto**] [**priv [aes-256]** *passphrase*] [**engineID** *id*] [**localizedkey**] [**localizedV2key**]] **Example:** `switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh`	Configures an SNMP user with authentication and privacy parameters. The passphrase can be any case-sensitive, alphanumeric string up to 64 characters. If you use the **localizedkey** keyword, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters. If the localizedV2key is used, the passphrase can be any case-sensitive, alphanumeric string upto 130 char without 0x/0X at the beginning. It is always preferred to collect localizedV2key from "show runn" since this is an encrypted data and cannot be generated offline. The engineID format is a 12-digit, colon-separated decimal number. **Note** • Beginning with Release 10.1(1), AES-128 is the default privacy protocol for SNMPv3.

	Command or Action	Purpose
		• Beginning with Cisco NX-OS Release 10.5(2), users can configure AES-256 as the privacy protocol for SNMPv3.
Step 3	(Optional) switch# **show snmp user** **Example:** `switch(config)# show snmp user`	Displays information about one or more SNMP users.
Step 4	(Optional) **copy running-config startup-config** **Example:** `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 IPv4 or IPv6 ACL to an SNMPv3 community to filter SNMP requests:

Command	Purpose
switch(config)# **snmp-server community** *name* [**use-ipv4acl** *ipv4acl-name*] [**use-ipv6acl** *ipv6acl-name*] `switch(config)# snmp-server community public use-ipv4acl myacl`	Assigns an IPv4 or IPv6 ACL to an SNMPv3 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch# **snmp-server host** *ip-address* **use-vrf** *vrf_name* [**udp_port** *number*]
3. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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.
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **snmp-server host** *ip-address* **filter-vrf** *vrf_name* [**udp_port** *number*]
3. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

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. 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.

SUMMARY STEPS

1. switch# **configuration terminal**
2. switch(config)# **snmp-server context** *context-name* **vrf** *vrf-name*

3. switch(config)# **snmp-server community** *community-name* **group** *group-name*
4. switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*

DETAILED STEPS

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. The names can be any alphanumeric string up to 32 characters. **Note** By default, SNMP sends the traps using the management VRF. If you do not want to use the management VRF, you must use this command to specify the desired VRF.
Step 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 20: Enabling SNMP Notifications

MIB	Related Commands
All notifications	**snmp-server enable traps**
BRIDGE-MIB	**snmp-server enable traps bridge newroot** **snmp-server enable traps bridge topologychange**
CISCO-AAA-SERVER-MIB	**snmp-server enable traps aaa**
ENITY-MIB, CISCO-ENTITY-FRU-CONTROL-MIB, CISCO-ENTITY-SENSOR-MIB	**snmp-server enable traps entity** **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** **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**
CISCO-FCS-MIB	**snmp-server enable traps fcs discovery-complete** **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**

MIB	Related Commands
CISCO-RSCN-MIB	**snmp-server enable traps rscn** **snmp-server enable traps rscn els** **snmp-server enable traps rscn ils**
CISCO-ZS-MIB	**snmp-server enable traps zone** **snmp-server enable traps zone default-zone-behavior-change** **snmp-server enable traps zone enhanced-zone-db-change** **snmp-server enable traps zone merge-failure** **snmp-server enable traps zone merge-success** **snmp-server enable traps zone request-reject** **snmp-server enable traps zone unsupp-mem**
CISCO-CONFIG-MAN-MIB **Note** 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.
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.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server enable traps link** [**cieLinkDown** | **cieLinkUp** | **cisco-xcvr-mon-status-chg** | **delayed-link-state-change**] | **extended-linkUp** | **extended-linkDown** | **linkDown** | **linkUp**]

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`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**] **Example:** `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).

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface** *type slot*/*port*
3. switch(config -if)# **no snmp trap link-status**

DETAILED STEPS

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.

SUMMARY STEPS

1. switch# **configuration terminal**
2. switch(config)# **snmp-server contact** *name*
3. switch(config)# **snmp-server location** *name*
4. (Optional) switch# **show snmp**
5. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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.

SUMMARY STEPS

1. switch# **configuration terminal**
2. switch(config)# **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]
3. switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*
4. (Optional) switch(config)# **no snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]

DETAILED STEPS

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. **Note** Do not enter an instance, VRF, or topology to delete a context mapping. If you use the **instance**, **vrf**, or **topology** keywords, you configure a mapping between the context and a zero-length string.

Disabling SNMP

SUMMARY STEPS

1. **configure terminal**
2. switch(config) # **no snmp-server protocol enable**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	switch(config) # **no snmp-server protocol enable** **Example:** `no snmp-server protocol enable`	Disables SNMP. SNMP is disabled by default.

Verifying the SNMP Configuration

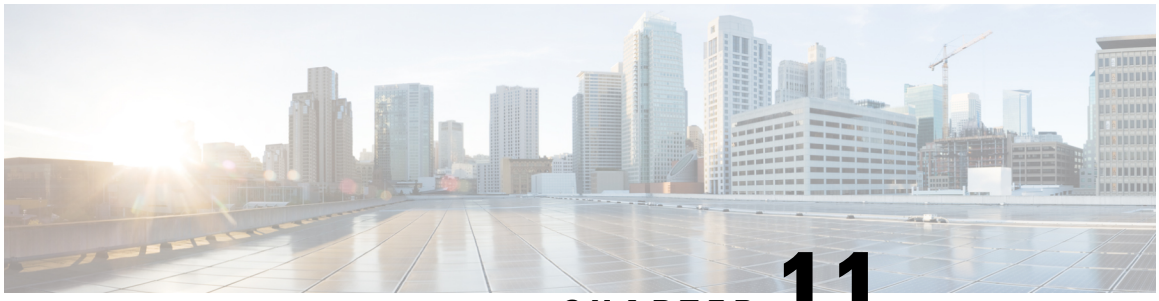
To display SNMP configuration information, perform one of the following tasks:

Command	Purpose
show snmp	Displays the SNMP status.
show snmp community	Displays the SNMP community strings.
show snmp engineID	Displays the SNMP engineID.
show snmp group	Displays SNMP roles.
show snmp sessions	Displays SNMP sessions.
show snmp trap	Displays the SNMP notifications enabled or disabled.
show snmp user	Displays SNMPv3 users.

Additional References

MIBs

MIBs	MIBs Link
MIBs related to SNMP	To locate and download supported MIBs, go to the following https://cisco.github.io/cisco-mibs/supportlists/nexus3548/Nexus3548MIBSupportList.html



CHAPTER 11

Configuring RMON

This chapter contains the following sections:

- Information About RMON, on page 133
- Configuration Guidelines and Limitations for RMON, on page 134
- Configuring RMON, on page 134
- Verifying the RMON Configuration, on page 137
- Default RMON Settings, on page 137

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.

Configuring RMON

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **rmon alarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-index*] **falling-threshold** *value* [*event-index*] [**owner** *name*]
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*]
4. (Optional) switch# **show rmon** {**alarms** | **hcalarms**}
5. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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.
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. 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **rmon event** *index* [**description** *string*] [**log**] [**trap**] [**owner** *name*]
3. (Optional) switch(config)# **show rmon** {**alarms** | **hcalarms**}
4. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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.

Verifying the RMON Configuration

Use the following commands to verify the RMON configuration information:

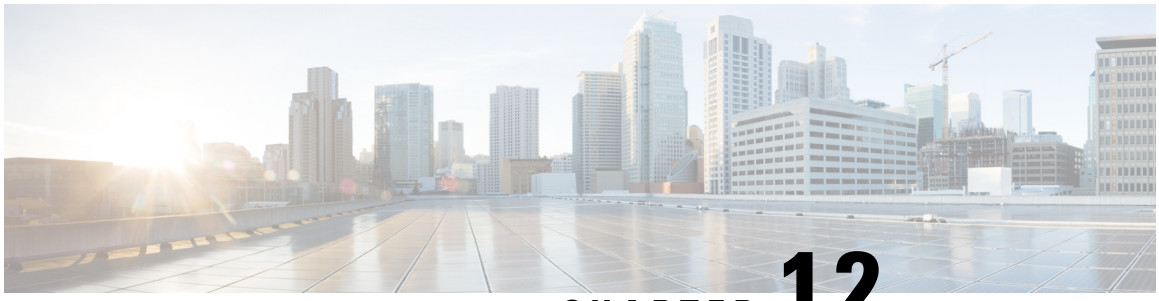
Command	Purpose
show rmon alarms	Displays information about RMON alarms.
show rmon events	Displays information about RMON events.
show rmon hcalarms	Displays information about RMON hcalarms.
show rmon logs	Displays information about RMON logs.

Default RMON Settings

The following table lists the default settings for RMON parameters.

Table 21: Default RMON Parameters

Parameters	Default
Alarms	None configured.
Events	None configured.



CHAPTER 12

Configuring Online Diagnostics

This chapter contains the following sections:

- Information About Online Diagnostics, on page 139
- Guidelines and Limitations for Online Diagnostics, on page 141
- Configuring Online Diagnostics, on page 141
- Verifying the Online Diagnostics Configuration, on page 142
- Default Settings for Online Diagnostics, on page 143

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 22: 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 23: 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.

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

Table 24: 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.
Front port	Tests the components (such as PHY and MAC) on the front ports.

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 25: 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 26: 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:

- Beginning from Cisco NX-OS Release 10.2(4), Backplane test is not supported on Nexus 3548 switches.

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **diagnostic bootup level** [**complete** | **bypass**]
3. (Optional) switch# **show diagnostic bootup level**

DETAILED STEPS

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: • **complete**—Performs all bootup diagnostics. This is the default value. • **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:

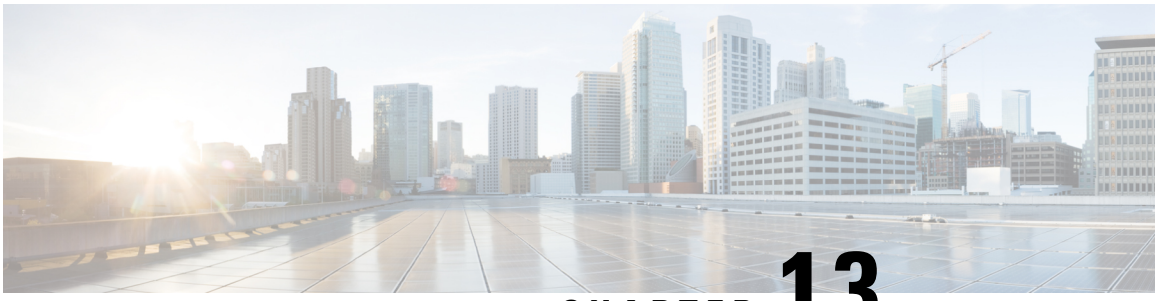
Command	Purpose
show diagnostic bootup level	Displays the bootup diagnostics level.
show diagnostic result module *slot*	Displays the results of the diagnostics tests.

Default Settings for Online Diagnostics

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

Table 27: Default Online Diagnostics Parameters

Parameters	Default
Bootup diagnostics level	complete



CHAPTER 13

Configuring Embedded Event Manager

This chapter contains the following sections:

- About Embedded Event Manager, on page 145
- Embedded Event Manager Policies, on page 146
- Prerequisites for Embedded Event Manager, on page 148
- Guidelines and Limitations for Embedded Event Manager, on page 148
- Default Settings for Embedded Event Manager, on page 149
- Defining an Environment Variable, on page 149
- Defining a User Policy Using the CLI, on page 150
- Configuring Event Statements, on page 152
- Configuring Action Statements, on page 154
- Defining a Policy Using a VSH Script, on page 156
- Registering and Activating a VSH Script Policy, on page 157
- Overriding a System Policy, on page 158
- Configuring Syslog as an EEM Publisher, on page 159
- Event Log Auto-Collection and Backup, on page 160

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.

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.
- 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.
- The following guidelines apply to Event Log Auto-Collection and Backup:
  - By default, enabled log collection on a switch provides between 15 minutes to several hours of event logs depending on size, scale and component activity.
  - To be able to collect relevant logs that span a longer period, only enable event log retention for the specific services/features you need. See "Enabling Extended Log File Retention For a Single Service". You can also export the internal event logs. See "External Log File Storage".
  - When troubleshooting, it is good practice to manually collect a snapshot of internal event logs in real time. See "Generating a Local Copy of Recent Log Files".
- 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.
- Cisco Nexus 3500 Series switches do not support Embedded Event Manager in Cisco NX-OS Release 7.0(3)I7(2) and the previous releases.

Default Settings for Embedded Event Manager

Table 28: Default EEM Parameters

Parameters	Default
System Policies	Active

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.

SUMMARY STEPS

1. **configure terminal**
2. **event manager environment** *variable-name variable-value*
3. (Optional) **show event manager environment** {*variable-name* | **all**}

4. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**event manager environment** *variable-name variable-value* **Example:** `switch(config) # event manager environment emailto "admin@anyplace.com"`	Creates an environment variable for EEM. The *variable-name* can be any case-sensitive, alphanumeric string up to 29 characters. 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**} **Example:** `switch(config) # show event manager environment all`	Displays information about the configured environment variables.
Step 4	(Optional) **copy running-config startup-config** **Example:** `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

SUMMARY STEPS

1. **configure terminal**
2. **event manager applet** *applet-name*
3. (Optional) **description** *policy-description*
4. **event** *event-statement*
5. (Optional) **tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | **or** {*tag*}] {**happens** *occurs* **in** *seconds*}
6. **action** *number*[.*number2*] *action-statement*
7. (Optional) **show event manager policy-state** *name* [**module** *module-id*]
8. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**event manager applet** *applet-name*
	Example:
	`switch(config)# event manager applet monitorShutdown`
	`switch(config-applet)#`
		Registers the applet with EEM and enters applet configuration mode.
		The applet-name can be any case-sensitive, alphanumeric string up to 29 characters.

Step 3	(Optional) **description** *policy-description* **Example:** `switch(config-applet)# description "Monitors interface shutdown."`	Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 4	**event** *event-statement* **Example:** `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*} **Example:** `switch(config-applet)# tag one or two happens 1 in 10000`	Correlates multiple events in the policy. The range for the *occurs* argument is from 1 to 4294967295. The range for the *seconds* argument is from 0 to 4294967295 seconds.
Step 6	**action** *number*[.*number2*] *action-statement* **Example:** `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*] **Example:** `switch(config-applet)# show event manager policy-state monitorShutdown`	Displays information about the status of the configured policy.
Step 8	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	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:

Before you begin

Define a user policy.

SUMMARY STEPS

1. **event cli** [**tag** *tag*] **match** *expression* [**count** *repeats* | **time** *seconds*
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**}
3. **event fanabsent** [**fan** *number*] **time** *seconds*
4. **event fanbad** [**fan** *number*] **time** *seconds*
5. **event memory** {**critical** | **minor** | **severe**}
6. **event policy-default count** *repeats* [**time** *seconds*]
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*
8. **event sysmgr memory** [**module** *module-num*] **major** *major-percent* **minor** *minor-percent* **clear** *clear-percent*
9. **event temperature** [**module** *slot*] [**sensor** *number*] **threshold** {**any** | **down** | **up**}
10. **event track** [**tag** *tag*] *object-number* **state** {**any** | **down** | **up**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**event cli** [**tag** *tag*] **match** *expression* [**count** *repeats* | **time** *seconds*	Triggers an event if you enter a command that matches the regular expression.
	Example: `switch(config-applet) # event cli match "shutdown"`	The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. The *repeats* range is from 1 to 65000. 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**}	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.

	Command or Action	Purpose
	Example: `switch(config-applet) # event counter name mycounter entry-val 20 gt`	The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. The *counter* name can be any case-sensitive, alphanumeric string up to 28 characters. The *entry* and *exit* value ranges are from 0 to 2147483647.
Step 3	**event fanabsent** [**fan** *number*] **time** *seconds* **Example:** `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. The *number* range is is from 1 to 1 and is module-dependent. The *seconds* range is from 10 to 64000.
Step 4	**event fanbad** [**fan** *number*] **time** *seconds* **Example:** `switch(config-applet) # event fanbad time 3000`	Triggers an event if a fan fails for more than the configured time, in seconds. The *number* range is module-dependent. The *seconds* range is from 10 to 64000.
Step 5	**event memory** {**critical** | **minor** | **severe**} **Example:** `switch(config-applet) # event memory critical`	Triggers an event if a memory threshold is crossed.
Step 6	**event policy-default count** *repeats* [**time** *seconds*] **Example:** `switch(config-applet) # event policy-default count 3`	Uses the event configured in the system policy. Use this option for overriding policies. The *repeats* range is from 1 to 65000. 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* **Example:** `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. The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. The *entry* and *exit* value ranges are from 0 to 18446744073709551615. The *time*, in seconds, is from 0 to 2147483647. 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* **Example:** `switch(config-applet) # event sysmgr memory minor 80`	Triggers an event if the specified system manager memory threshold is exceeded. The *percent* range is from 1 to 99.

	Command or Action	Purpose
Step 9	**event temperature** [**module** *slot*] [**sensor** *number*] **threshold** {**any** | **down** | **up**} **Example:** `switch(config-applet) # event temperature module 2 threshold any`	Triggers an event if the temperature sensor exceeds the configured threshold. The *sensor* range is from 1 to 18.
Step 10	**event track** [**tag** *tag*] *object-number* **state** {**any** | **down** | **up** **Example:** `switch(config-applet) # event track 1 state down`	Triggers an event if the tracked object is in the configured state. The **tag** *tag* keyword-argument pair identifies this specific event when multiple events are included in the policy. 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.

SUMMARY STEPS

1. **action** *number*[.*number2*] **cli** *command1*[*command2.*] [**local**]
2. **action** *number*[.*number2*] **counter name** *counter* **value** *val* **op** {**dec** | **inc** | **nop** | **set**}
3. **action** *number*[.*number2*] **event-default**

4. **action** *number*[.*number2*] **policy-default**
5. **action** *number*[.*number2*] **reload** [**module** *slot* [**-** *slot*]]
6. **action** *number*[.*number2*] **snmp-trap** [**intdata1** *integer-data1*] [**intdata2** *integer-data2*] [**strdata** *string-data*]
7. **action** *number*[.*number2*] **syslog** [**priority** *prio-val*] **msg** *error-message*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**action** *number*[.*number2*] **cli** *command1*[*command2.*] [**local**] **Example:** `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. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9.
Step 2	**action** *number*[.*number2*] **counter name** *counter* **value** *val* **op** {**dec** | **inc** | **nop** | **set**} **Example:** `switch(config-applet) # action 2.0 counter name mycounter value 20 op inc`	Modifies the counter by the configured value and operation. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9. The *counter* can be any case-sensitive, alphanumeric string up to 28 characters. The *val* can be an integer from 0 to 2147483647 or a substituted parameter.
Step 3	**action** *number*[.*number2*] **event-default** **Example:** `switch(config-applet) # action 1.0 event-default`	Completes the default action for the associated event. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9.
Step 4	**action** *number*[.*number2*] **policy-default** **Example:** `switch(config-applet) # action 1.0 policy-default`	Completes the default action for the policy that you are overriding. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9.
Step 5	**action** *number*[.*number2*] **reload** [**module** *slot* [**-** *slot*]] **Example:** `switch(config-applet) # action 1.0 reload module 3-5`	Forces one or more modules to the entire system to reload. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9.

	Command or Action	Purpose
Step 6	**action** *number*[.*number2*] **snmp-trap** [**intdata1** *integer-data1*] [**intdata2** *integer-data2*] [**strdata** *string-data*] **Example:** `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. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9. The *data* elements can be any number up to 80 digits. The *string* can be any alphanumeric string up to 80 characters.
Step 7	**action** *number*[.*number2*] **syslog** [**priority** *prio-val*] **msg** *error-message* **Example:** `switch(config-applet) # action 1.0 syslog priority notifications msg "cpu high"`	Sends a customized syslog message at the configured priority. The action label is in the format number1.number2. The *number* can be any number from 1 to 16 digits. The range for *number2* is from 0 to 9. 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:

SUMMARY STEPS

1. In a text editor, list the commands that define the policy.
2. Name the text file and save it.
3. Copy the file to the following system directory: bootflash://eem/user_script_policies

DETAILED STEPS

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.

SUMMARY STEPS

1. **configure terminal**
2. **event manager policy** *policy-script*
3. (Optional) **event manager policy internal** *name*
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**event manager policy** *policy-script* **Example:** `switch(config)# event manager policy moduleScript`	Registers and activates an EEM script policy. The *policy-script* can be any case-sensitive, alphanumeric string up to 29 characters.
Step 3	(Optional) **event manager policy internal** *name* **Example:**	Registers and activates an EEM script policy. The *policy-script* can be any case-sensitive alphanumeric string up to 29 characters.

	Command or Action	Purpose
	`switch(config)# event manager policy internal moduleScript`	
Step 4	(Optional) **copy running-config startup-config** **Example:** `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

SUMMARY STEPS

1. **configure terminal**
2. (Optional) **show event manager policy-state** *system-policy*
3. **event manager applet** *applet-name* **override** *system-policy*
4. **description** *policy-description*
5. **event** *event-statement*
6. **section** *number action-statement*
7. (Optional) **show event manager policy-state** *name*
8. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	(Optional) **show event manager policy-state** *system-policy*
	Example:
	`switch(config-applet)# show event manager policy-state __ethpm_link_flap`
	`Policy __ethpm_link_flap`
		Displays information about the system policy that you want to override, including thresholds. Use the **show event manager system-policy** command to find the system policy names.

	Command or Action	Purpose
	Cfg count : 5 Cfg time interval : 10.000000 (seconds) Hash default, Count 0	

Step 3	**event manager applet** *applet-name* **override** *system-policy*
	Example:
	`switch(config-applet)# event manager applet ethport override __ethpm_link_flap`
	`switch(config-applet)#`
		Overrides a system policy and enters applet configuration mode.
		The *applet-name* can be any case-sensitive, alphanumeric string up to 80 characters.
		The *system-policy* must be one of the system policies.

Step 4	**description** *policy-description* **Example:** `switch(config-applet)# description "Overrides link flap policy"`	Configures a descriptive string for the policy. 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* **Example:** `switch(config-applet)# event policy-default count 2 time 1000`	Configures the event statement for the policy.
Step 6	**section** *number action-statement* **Example:** `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* **Example:** `switch(config-applet)# show event manager policy-state ethport`	Displays information about the configured policy.
Step 8	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	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.

SUMMARY STEPS

1. **configure terminal**
2. **event manager applet** *applet-name*
3. **event syslog** [**tag** *tag*] {**occurs** *number* | **period** *seconds* | **pattern** *msg-text* | **priority** *priority*}
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**event manager applet** *applet-name*
	Example:
	`switch(config)# event manager applet abc`
	`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*} **Example:** `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** **Example:** `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.

Event Log Auto-Collection and Backup

Automatically collected event logs are stored locally on switch memory. Event log file storage is a temporary buffer that stores files for a fixed amount of time. Once the time period has elapsed, a roll-over of the buffer makes room for the next files. The roll-over uses a first-in-first-out method.

Beginning with Cisco NX-OS Release 9.3(3), EEM uses the following methods of collection and backup:

- Extended Log File Retention
- Trigger-Based Event Log Auto-Collection

Extended Log File Retention

Beginning with Cisco NX-OS release 9.3(3), all Cisco Nexus platform switches, with at least 8Gb of system memory, support the extended retention of event logging files. Storing the log files locally on the switch or remotely through an external container, reduces the loss of event logs due to rollover.

Enabling Extended Log File Retention For All Services

Extended Log File Retention is enabled by default for all services running on a switch. If the switch doesn't have the log file retention feature enabled (**no bloggerd log-dump** is configured), use the following procedure to enable it.

SUMMARY STEPS

1. **configure terminal**
2. **bloggerd log-dump all**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**bloggerd log-dump all**
	Example:
	`switch(config)# bloggerd log-dump all`
	`switch(config)#`
		Enables the log file retention feature for all services.

Example

```
switch# configure terminal
switch(config)# bloggerd log-dump all
Sending Enable Request to Bloggerd
Bloggerd Log Dump Successfully enabled
switch(config)#
```

Disabling Extended Log File Retention For All Services

Extended Log File Retention is disabled by default for all services on the switch. If the switch has the log file retention feature enabled for all services and you want to disable it, use the following procedure.

SUMMARY STEPS

1. **configure terminal**
2. **no bloggerd log-dump all**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**no bloggerd log-dump all**
	Example:
	`switch(config)# no bloggerd log-dump all`
	`switch(config)#`
		Disables the log file retention feature for all services on the switch.

Example

```
switch# configure terminal
switch(config)# no bloggerd log-dump all
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)#
```

Enabling Extended Log File Retention For a Single Service

Extended Log File Retention is enabled by default for all services running on a switch. If the switch doesn't have the log file retention feature enabled (**no bloggerd log-dump** is configured), use the following procedure to enable it for a single service.

SUMMARY STEPS

1. **show system internal sysmgr service name** *service-type*
2. **configure terminal**
3. **bloggerd log-dump sap** *number*
4. **show system internal bloggerd info log-dump-info**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**show system internal sysmgr service name** *service-type* **Example:** `switch# show system internal sysmgr service name aclmgr`	Displays information about the ACL Manager including the service SAP number.

Step 2	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 3	**bloggerd log-dump sap** *number* **Example:** `switch(config)# bloggerd log-dump sap 351`	Enables the log file retention feature for the ACL Manager service.
Step 4	**show system internal bloggerd info log-dump-info** **Example:** `switch(config)# show system internal bloggerd info log-dump-info`	Displays information about the log file retention feature on the switch.

Example

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
        UUID = 0x182, PID = 653, SAP = 351
        State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
        Restart count: 1
        Time of last restart: Mon Nov  4 11:10:39 2019.
        The service never crashed since the last reboot.
        Tag = N/A
        Plugin ID: 0
switch(config)# configure terminal
switch(config)# bloggerd log-dump sap 351
Sending Enable Request to Bloggerd
Bloggerd Log Dump Successfully enabled
switch(config)# show system internal bloggerd info log-dump-info
------------------------------------------------------------
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
------------------------------------------------------------
Module     | VDC        | SAP                       | Enabled?
------------------------------------------------------------
1          | 1          | 351 (MTS_SAP_ACLMGR     ) | Enabled
------------------------------------------------------------
------------------------------------------------------------
Log Dump Throttle Switch-Wide Config:
-------------------------------------
```

```
Log Dump Throttle                                       : ENABLED
Minimum buffer rollover count (before throttling)       : 5
Maximum allowed rollover count per minute               : 1
----------------------------------------------------------------

switch(config)#
```

Displaying Extended Log Files

Use this task to display the event log files currently stored on the switch.

SUMMARY STEPS

1. **dir debug:log-dump/**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**dir debug:log-dump/** **Example:** `switch# dir debug:log-dump/`	Displays the event log files currently stored on the switch.

Example

```
switch# dir debug:log-dump/

3676160 Dec 05 02:43:01 2019 20191205023755_evtlog_archive.tar
3553280 Dec 05 06:05:06 2019 20191205060005_evtlog_archive.tar

Usage for debug://sup-local
913408 bytes used
4329472 bytes free
5242880 bytes total
```

Disabling Extended Log File Retention For a Single Service

Extended Log File Retention is enabled by default for all services on the switch. If the switch has the log file retention feature enabled for a single service or all services (by default in Cisco NX-OS Release 9.3(5)), and you want to disable a specific service or services, use the following procedure.

SUMMARY STEPS

1. **show system internal sysmgr service name** *service-type*
2. **configure terminal**
3. **no bloggerd log-dump sap** *number*
4. **show system internal bloggerd info log-dump-info**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**show system internal sysmgr service name** *service-type* **Example:** `switch# show system internal sysmgr service name aclmgr`	Displays information about the ACL Manager including the service SAP number.

Step 2	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 3	**no bloggerd log-dump sap** *number* **Example:** `switch(config)# no bloggerd log-dump sap 351`	Disables the log file retention feature for the ACL Manager service.
Step 4	**show system internal bloggerd info log-dump-info** **Example:** `switch(config)# show system internal bloggerd info log-dump-info`	Displays information about the log file retention feature on the switch.

Example

The following example shows how to disable extended log file retention for a service named "aclmgr":

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
        UUID = 0x182, PID = 653, SAP = 351
        State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
        Restart count: 1
        Time of last restart: Mon Nov  4 11:10:39 2019.
        The service never crashed since the last reboot.
        Tag = N/A
        Plugin ID: 0
switch(config)# configure terminal
switch(config)# no bloggerd log-dump sap 351
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)# show system internal bloggerd info log-dump-info
------------------------------------------------------------
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
------------------------------------------------------------
Module     | VDC        | SAP                       | Enabled?
------------------------------------------------------------
1          | 1          | 351 (MTS_SAP_ACLMGR     ) | Disabled
------------------------------------------------------------
------------------------------------------------------------
```

```
Log Dump Throttle Switch-Wide Config:
-------------------------------------
Log Dump Throttle                                    : ENABLED
Minimum buffer rollover count (before throttling)    : 5
Maximum allowed rollover count per minute            : 1
-------------------------------------------------------------

switch(config)#
```

Trigger-Based Event Log Auto-Collection

Trigger-based log collection capabilities:

- Automatically collect relevant data when issues occur.
- No impact on control plane
- Customizable configuration:
  - Defaults populated by Cisco
  - Selectively override what-to-collect by network administrator or by Cisco TAC.
  - Automatically update new triggers on image upgrades.
- Store logs locally on the switch or remotely on an external server.
- Supports severity 0, 1, and 2 syslogs:
- Custom syslogs for ad-hoc events (auto-collection commands attached to the syslogs)

Enabling Trigger-Based Log File Auto-Collection

To enable trigger-based automatic creation of log files, you must create an override policy for the __syslog_trigger_default system policy with a custom YAML file and define the specific logs for which information will be collected.

For more information on creating a custom YAML file to enable log file auto-collection, see Configuring the Auto-Collection YAML File, on page 167.

Auto-Collection YAML File

The Auto-Collection YAML file that is specified in the **action** command in the EEM function, defines actions for different system or feature components. This file is located in the switch directory: /bootflash/scripts. In addition to the default YAML file, you can create component-specific YAML files and place them in the same directory. The naming convention for component-specific YAML files is **component-name.yaml**. If a component-specific file is present in the same directory, it takes precedence over the file that is specified in the **action** command. For example, if the action file, **`bootflash/scripts/platform.yaml`** is in the **`/bootflash/scripts`** directory with the default action file, **`bootflash/scripts/test.yaml`**, then the instructions defined in **`platform.yaml`** file take precedence over the instructions for the platform component present in the default **`test.yaml`** file.

Examples of components are, ARP, BGP, IS-IS, and so on. If you are not familiar with all the component names, contact Cisco Customer Support for assistance in defining the YAML file for component-specific actions (and for the default **test.yaml** file as well).

Example:

```
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml $_syslog_msg
```

Configuring the Auto-Collection YAML File

A contents of a YAML file determines the data collected during trigger-based auto-collection. There must be only one YAML file on the switch but it can contain auto-collection meta-data for any number of switch components and messages.

Locate the YAML file in the following directory on the switch:

```
/bootflash/scripts
```

Invoke the YAML file for trigger-based collection by using the following example. The example shows the minimum required configuration for trigger-based collection to work with a user-defined YAML file.

```
switch# show running-config eem
!Command: show running-config eem
!Running configuration last done at: Mon Sep 30 19:34:54 2019
!Time: Mon Sep 30 22:24:55 2019
version 9.3(3) Bios:version 07.59
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml $_syslog_msg
```

In the preceding example, "`test_1`" is the name of the applet and "`test.yaml`" is the name of the user-configured YAML file present in the /bootflash/scripts directory.

Example YAML File

The following is an example of a basic YAML file supporting the trigger-based event log auto-collection feature. The definitions for the keys/values in the file are in the table that follows.



Note Make sure that the YMAL file has proper indentation. As a best practice, run it through any "online YAML validator" before using it on a switch.

```
bash-4.3$ cat /bootflash/scripts/test.yaml
version: 1
components:
   securityd:
        default:
            tech-sup: port
            commands: show module
   platform:
        default:
            tech-sup: port
            commands: show module
```

Key: Value	Description
version: 1	Set to 1. Any other number creates an incompatibility for the auto collect script.
components:	Keyword specifying that what follows are switch components.
securityd:	Name of the syslog component (`securityd` is a facility name in syslog).
default:	Identifies all messages belonging to the component.

Key: Value	Description
tech-sup: port	Collect tech support of the port module for the `securityd` syslog component.
commands: show module	Collect show module command output for the `securityd` syslog component.
platform:	Name of the syslog component (`platform` is a facility name in syslog).
tech-sup: port	Collect tech support of the port module for the `platform` syslog component.
commands: show module	Collect show module command output for the `platform` syslog component.

Use the following example to associate auto-collect metadata only for a specific log. For example, SECURITYD-2-FEATURE_ENABLE_DISABLE

```
securityd:
                feature_enable_disable:
                        tech-sup: security
                        commands: show module
```

Key: Value	Description
securityd:	Name of the syslog component (`securityd` is a facility name in syslog).
feature_enable_disable:	Message ID of the syslog message.
tech-sup: security	Collect tech support of the security module for the `securityd` syslog component.
commands: show module	Collect show module command output for the security syslog component.

Example syslog output for the above YAML entry:

```
2019 Dec 4 12:41:01 n9k-c93108tc-fx %SECURITYD-2-FEATURE_ENABLE_DISABLE: User
has enabled the feature bash-shell
```

Use the following example to specify multiple values.

```
version: 1
components:
   securityd:
        default:
            commands: show module;show version;show module
            tech-sup: port;lldp
```



Note Use semicolons to separate multiple show commands and tech support key values (see the preceding example).

Beginning with Release 10.1(1), `test.yaml` can be replaced with a folder inside which more than one YAML files can be present. All the YAML files in the folder must follow the `ComponentName.yaml` naming convention.

In the following example, `test.yaml` is replaced with `test_folder`:

```
test.yaml:
```

```
event manager applet logging2 override __syslog_trigger_default
   action 1.0 collect test.yaml rate-limt 30 $_syslog_msg

test_folder:
event manager applet logging2 override __syslog_trigger_default
   action 1.0 collect test_folder rate-limt 30 $_syslog_msg
```

The following example shows the path and component(s) for `test_folder`:

```
ls /bootflash/scripts/test_folder
bgp.yaml ppm.yaml
```

Limiting the Amount of Auto-Collections Per Component

For auto-collection, the limit of the number of bundles per component event is set to three (3) by default. If more than three events occur for a component, then the events are dropped with the status message **`EVENTLOGLIMITREACHED`**. The auto-collection of the component event restarts when the event log has rolled over.

Example:

```
switch# show system internal event-logs auto-collect history
DateTime             Snapshot ID  Syslog                  Status/Secs/Logsize(Bytes)
2020-Jun-27 07:20:03  1140276903   ACLMGR-0-TEST_SYSLOG     EVENTLOGLIMITREACHED
2020-Jun-27 07:15:14  1026359228   ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:15:09  384952880    ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:13:55  1679333688   ACLMGR-0-TEST_SYSLOG     PROCESSED:2:9332278
2020-Jun-27 07:13:52  1679333688   ACLMGR-0-TEST_SYSLOG     PROCESSING
2020-Jun-27 07:12:55  502545693    ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:12:25  1718497217   ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:08:25  1432687513   ACLMGR-0-TEST_SYSLOG     PROCESSED:2:10453823
2020-Jun-27 07:08:22  1432687513   ACLMGR-0-TEST_SYSLOG     PROCESSING
2020-Jun-27 07:06:16  90042807     ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:03:26  1737578642   ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:02:56  40101277     ACLMGR-0-TEST_SYSLOG     PROCESSED:3:10542045
2020-Jun-27 07:02:52  40101277     ACLMGR-0-TEST_SYSLOG     PROCESSING
```

Auto-Collection Log Files

About Auto-Collection Log Files

The configuration in a YAML file determines the contents of an auto-collected log file. You can't configure the amount of memory used for collected log files. You can configure the frequency of when the stored files get purged.

Autocollected log files get saved in the following directory:

```
switch# dir bootflash:eem_snapshots
   44205843    Sep 25 11:08:04 2019
1480625546_SECURITYD_2_FEATURE_ENABLE_DISABLE_eem_snapshot.tar.gz
  Usage for bootflash://sup-local
 6940545024 bytes used
44829761536 bytes free
51770306560 bytes total
```

Accessing the Log Files

Locate the logs by using the command keyword "`debug`":

```
switch# dir debug:///
...
       26    Oct 22 10:46:31 2019  log-dump
       24    Oct 22 10:46:31 2019  log-snapshot-auto
       26    Oct 22 10:46:31 2019  log-snapshot-user
```

The following table describes the log locations and the log types stored.

Location	Description
log-dump	This folder stores Event logs on log rollover.
log-snapshot-auto	This folder contains the auto-collected logs for syslog events 0, 1, 2.
log-snapshot-user	This folder stores the collected logs when you run the `bloggerd log-snapshot <>` command.

Use the following example to view the log files generated on log rollover:

```
switch# dir debug:log-dump/
debug:log-dump/20191022104656_evtlog_archive.tar
debug:log-dump/20191022111241_evtlog_archive.tar
debug:log-dump/20191022111841_evtlog_archive.tar
debug:log-dump/20191022112431_evtlog_archive.tar
debug:log-dump/20191022113042_evtlog_archive.tar
debug:log-dump/20191022113603_evtlog_archive.tar
```

Parsing the Log tar Files

Use the following example to parse the logs in the tar files:

```
switch# show system internal event-logs parse debug:log-dump/20191022104656_evtlog_archive.tar
--------LOGS:/tmp/BLOGGERD0.991453012199/tmp/1-191022104658-191022110741-device_test-M27-V1-I1:0-P884.gz--------
2019 Oct 22 11:07:41.597864 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Data Space
Limits(bytes): Soft: -1  Ha rd: -1
2019 Oct 22 11:07:41.597857 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Stack Space
Limits(bytes): Soft: 500000  Hard: 500000
2019 Oct 22 11:07:41.597850 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):AS: 1005952076
-1
2019 Oct 22 11:07:41.597406 E_DEBUG Oct 22 11:07:41 2019(device_test_process_events):Sdwrap
 msg unknown
2019 Oct 22 11:07:41.597398 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Going back to
select
2019 Oct 22 11:07:41.597395 E_DEBUG Oct 22 11:07:41 2019(nvram_test):TestNvram examine 27
blocks
2019 Oct 22 11:07:41.597371 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Parent: Thread
created test index:4 thread_id:-707265728
2019 Oct 22 11:07:41.597333 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Node inserted
2019 Oct 22 11:07:41.597328 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):The test index
in diag is 4
2019 Oct 22 11:07:41.597322 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):result severity
level
2019 Oct 22 11:07:41.597316 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):callhome alert
level
```

The following table describes the additional keywords available for parsing the specific tar file:

Keyword	Description
component	Decode logs belonging to the component identified by process name.
from-datetime	Decode logs from a specific date and time in yy[mm[dd[HH[MM[SS]]]]] format.

Keyword	Description
instance	List of SDWRAP buffer instances to be decoded (comma separated).
module	Decode logs from modules such as SUP and LC (using module IDs).
to-datetime	Decode logs up to a specific date and time in yy[mm[dd[HH[MM[SS]]]]] format.

Copying Logs to a Different Location

Use the following example to copy logs to a different location such as a remote server:

```
switch# copy debug:log-dump/20191022104656_evtlog_archive.tar
scp://<ip-adress>/nobackup/<user> vrf management  use-kstack
Enter username: user@<ip-address>'s password:
20191022104656_evtlog_archive.tar                                                 100%  130KB
130.0KB/s   00:00
Copy complete, now saving to disk (please wait)...
Copy complete.
```

Purging Auto-Collection Log Files

There are two types of generated trigger-based auto-collection logs: EventHistory and EventBundle.

Purge Logic for EventHistory Logs

For event history, purging occurs in the /var/sysmgr/srv_logs/xport folder. 250MB of partitioned RAM is mounted at /var/sysmgr/srv_logs directory.

If the /var/sysmgr/srv_logs memory usage is under 65% of the 250MB allocated, no files get purged. When the memory utilization reaches the 65% limit level, the oldest files get purged until there's enough memory available to continue saving new logs.

Purge Logic for EventBundle Logs

For event bundles, the purge logic occurs in the /bootflash/eem_snapshots folder. For storing the auto-collected snapshots, the EEM auto-collect script allocates 5% of the bootflash storage. The logs get purged once the 5% bootflash capacity is used.

When a new auto-collected log is available but there's no space to save it in bootflash (already at 5% capacity), the system checks the following:

1. If there are existing auto-collected files that are more than 12 hours old, the system deletes the files and the new logs get copied.
2. If the existing auto collected files are less than 12 hours old, the system discards the newly collected logs without saving them.

You can modify the 12-hour default purge time by using the following commands. The time specified in the command is in minutes.

```
switch(config)# event manager applet test override __syslog_trigger_default
switch(config-applet)# action 1.0 collect test.yaml purge-time 300 $_syslog_msg
```

event manager command: *test* is an example name for the policy. **__syslog_trigger_default** is the name of the system policy that you want to override. This name must begin with a double underscore (__).

action command: **1.0** is an example number for the order in which the action is executed. **collect** indicates that data is collected using the YAML file. *test.yaml* is an example name of the YAML file. **$_syslog_msg** is the name of the component.



Note At any given time, there can be only one trigger-based auto-collection event in progress. If another new log event is attempting to be stored when auto-collection is already occurring, the new log event is discarded.

By default, there's only one trigger-based bundle collected every five minutes (300 sec). This rate limiting is also configurable by the following commands. The time specified in the command is in seconds.

```
switch(config)# event manager applet test override __syslog_trigger_default
switch(config-applet)# action 1.0 collect test.yaml rate-limit 600 $_syslog_msg
```

event manager command: *test* is an example name for the policy. **__syslog_trigger_default** is an example name of the system policy to override. This name must begin with a double underscore (__).

action command: **1.0** is an example number for the order in which the action is executed. **collect** indicates that data is collected using the YAML file. *test.yaml* is an example name of the YAML file. **$_syslog_msg** is the name of the component.

Beginning with Release 10.1(1), the rate of collection can also be regulated using a maximum number of triggers option, ensuring that only those many number of triggers are honored. After the **max-triggers** value is reached, no more bundles will be collected on the syslog occurrence.

```
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml rate-limt 30 max-triggers 5 $_syslog_msg
```



Note If you delete auto collected bundles manually from `debug:log-snapshot-auto/`, then it will restart the collection based on the configured number of **max-triggers** when the next event occurs.

Auto-Collection Statistics and History

The following example shows trigger-based collection statistics:

```
switch# show system internal event-logs auto-collect statistics
--------------------EEM Auto Collection Statistics--------------------
Syslog Parse Successful :88 Syslog Parse Failure :0
Syslog Ratelimited :0 Rate Limit Check Failed :0
Syslog Dropped(Last Action In Prog) :53 Storage Limit Reached :0
User Yaml Action File Unavailable :0 User Yaml Parse Successful :35
User Yaml Parse Error :0 Sys Yaml Action File Unavailable :11
Sys Yaml Parse Successful :3 Sys Yaml Parse Error :0
Yaml Action Not Defined :0 Syslog Processing Initiated :24
Log Collection Failed :0 Tar Creation Error :0
Signal Interrupt :0 Script Exception :0
Syslog Processed Successfully :24 Logfiles Purged :0
```

The following example shows trigger-based collection history (the processed syslogs, process time, size of the data collected) obtained using a CLI command:

```
switch# show system internal event-logs auto-collect history
DateTime Snapshot ID Syslog Status/Secs/Logsize(Bytes)
2019-Dec-04 05:30:32 1310232084 VPC-0-TEST_SYSLOG PROCESSED:9:22312929
2019-Dec-04 05:30:22 1310232084 VPC-0-TEST_SYSLOG PROCESSING
2019-Dec-04 04:30:13 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSED:173:33194665
```

```
2019-Dec-04 04:28:47 897805674 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:28:47 947981421 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:27:19 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSING
2019-Dec-04 02:17:16 1957148102 CARDCLIENT-2-FPGA_BOOT_GOLDEN NOYAMLFILEFOUND
```

Verifying Trigger-Based Log Collection

Verify that the trigger-based log collection feature is enabled by entering the **show event manager system-policy | i trigger** command as in this example:

```
switch# show event manager system-policy | i trigger n 2
           Name : __syslog_trigger_default
    Description : Default policy for trigger based logging
    Overridable : Yes
     Event type : 0x2101
```

Checking Trigger-Based Log File Generation

You can check to see if the trigger-based auto-collection feature has generated any event log files. Enter one of the commands in the following examples:

```
switch# dir bootflash:eem_snapshots
9162547 Nov 12 22:33:15 2019 1006309316_SECURITYD_2_FEATURE_ENABLE_DISABLE_eem_snapshot.tar.gz

Usage for bootflash://sup-local
8911929344 bytes used
3555950592 bytes free
12467879936 bytes total
```

```
switch# dir debug:log-snapshot-auto/
63435992 Dec 03 06:28:52 2019
20191203062841_1394408030_PLATFORM_2_MOD_PWRDN_eem_snapshot.tar.gz

Usage for debug://sup-local
544768 bytes used
4698112 bytes free
5242880 bytes total
```

Local Log File Storage

Local log file storage capabilities:

- Amount of local data storage time depends on the scale, and type, of deployment. For both modular and nonmodular switches, the storage time is from 15 minutes to several hours of data. To be able to collect relevant logs that span a longer period:
  - Only enable event log retention for the specific services/features you need. See Enabling Extended Log File Retention For a Single Service , on page 162.
  - Export the internal event logs off the switch. See External Log File Storage, on page 176.
- Compressed logs are stored in RAM.
- 250MB memory is reserved for log file storage.
- Log files are optimized in tar format (one file for every five minutes or 10MB, whichever occurs first).
- Allow snap-shot collection.

Generating a Local Copy of Recent Log Files

Extended Log File Retention is enabled by default for all services running on a switch. For local storage, the log files are stored on flash memory. Use the following procedure to generate a copy of up to ten of the most recent event log files.

SUMMARY STEPS

1. **bloggerd log-snapshot** [*file-name*] [**bootflash:** *file-path* | **logflash:** *file-path* | **usb1:**] [**size** *file-size*] [**time** *minutes*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**bloggerd log-snapshot** [*file-name*] [**bootflash:** *file-path* | **logflash:** *file-path* | **usb1:**] [**size** *file-size*] [**time** *minutes*] **Example:** `switch# bloggerd log-snapshot snapshot1`	Creates a snapshot bundle file of the last ten event logs stored on the switch. Default storage for this operation is **logflash**. *file-name*: The filename of the generated snapshot log file bundle. Use a maximum of 64 characters for *file-name*. **Note** This variable is optional. If it is not configured, the system applies a timestamp and "_snapshot_bundle.tar" as the filename. Example: `20200605161704_snapshot_bundle.tar` **bootflash:** *file-path*: The file path where the snapshot log file bundle is being stored on the bootflash. Choose one of the following initial paths: • bootflash:/// • bootflash://module-1/ • bootflash://sup-1/ • bootflash://sup-active/ • bootflash://sup-local/ **logflash:** *file-path*: The file path where the snapshot log file bundle is being stored on the logflash. Choose one of the following initial paths: • logflash:/// • logflash://module-1/ • logflash://sup-1/ • logflash://sup-active/

	Command or Action	Purpose
		• logflash://sup-local/ **usb1:**: The file path where the snapshot log file bundle is being stored on the USB device. **size** *file-size*: The snapshot log file bundle based on size in megabytes (MB). Range is from 5MB through 250MB. **time** *minutes*: The snapshot log file bundle based on the last x amount of time (minutes). Range is from 1 minute through 30 minutes.

Example

```
switch# bloggerd log-snapshot snapshot1
Snapshot generated at logflash:evt_log_snapshot/snapshot1_snapshot_bundle.tar Please cleanup
 once done.
switch#
switch# dir logflash:evt_log_snapshot
159098880 Dec 05 06:40:24 2019 snapshot1_snapshot_bundle.tar
159354880 Dec 05 06:40:40 2019 snapshot2_snapshot_bundle.tar

Usage for logflash://sup-local
759865344 bytes used
5697142784 bytes free
6457008128 bytes total
```

Display the same files using the command in this example:

```
switch# dir debug:log-snapshot-user/
159098880 Dec 05 06:40:24 2019 snapshot1_snapshot_bundle.tar
159354880 Dec 05 06:40:40 2019 snapshot2_snapshot_bundle.tar

Usage for debug://sup-local
929792 bytes used
4313088 bytes free
5242880 bytes total
```



Note The file name is identified at the end of the example. Each individual log file is also identified by the date and time it was generated.

Beginning with Release 10.1(1), the LC core file includes the `log-snapshot` bundle. The `log-snapshot` bundle filename is `tac_snapshot_bundle.tar.gz`. An example is shown below:

```
bash-4.2$ tar -tvf 1610003655_0x102_aclqos_log.17194.tar.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 pss/
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_runtime_info_lc.gz
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_runtime_cfg_lc.gz
-rw-rw-rw- root/root 107 2021-01-07 12:44 pss/dev_shm_aclqos_debug.gz
-rw-rw-rw- root/root 129583 2021-01-07 12:44 pss/clqosdb_ver1_0_user.gz
-rw-rw-rw- root/root 20291 2021-01-07 12:44 pss/clqosdb_ver1_0_node.gz
-rw-rw-rw- root/root 444 2021-01-07 12:44 pss/clqosdb_ver1_0_ctrl.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 proc/
-rw-rw-rw- root/root 15159 2021-01-07 12:44 0x102_aclqos_compress.17194.log.25162
-rw-rw-rw- root/root 9172392 2021-01-07 12:43 0x102_aclqos_core.17194.gz
```

```
-rw-rw-rw- root/root 43878 2021-01-07 12:44 0x102_aclqos_df_dmesg.17194.log.gz
-rw-rw-rw- root/root 93 2021-01-07 12:44 0x102_aclqos_log.17194
-rw-rw-rw- root/root 158 2021-01-07 12:44 0x102_aclqos_mcore.17194.log.gz
drwxrwxrwx root/root 0 2021-01-07 12:44 usd17194/
-rw-rw-rw- root/root 11374171 2021-01-07 12:44 tac_snapshot_bundle.tar.gz
```

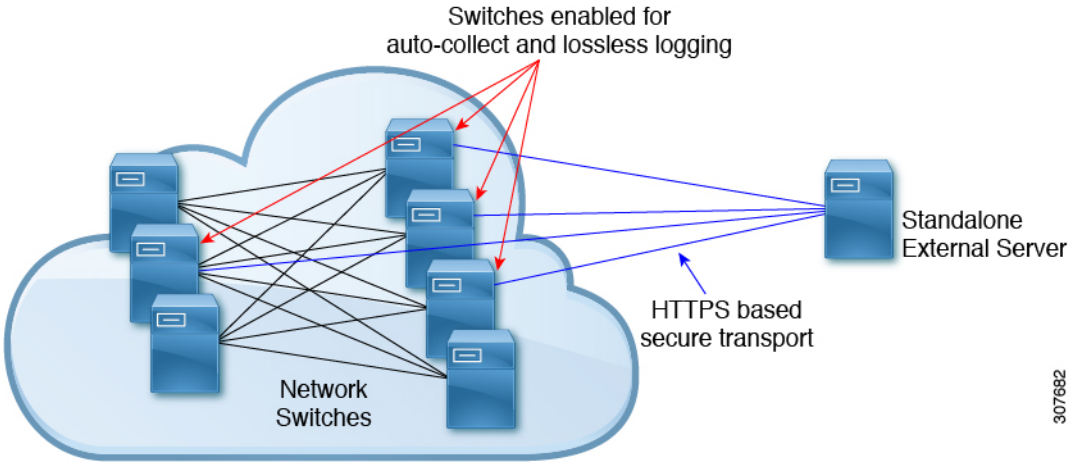
External Log File Storage

An external server solution provides the capability to store logs off-switch in a secure manner.

Note To create the external storage capability, contact Cisco Technical Assistance Center(TAC) to help deploy the external server solution.

Te following are external log file storage capabilities:

- Enabled on-demand
- HTTPS-based transport
- Storage requirements:
  - Nonmodular switches: 300MB
  - Modular switches: 12GB (per day, per switch)
- An external server generally stores logs for 10 switches. However, there's no firm limit to the number of switches supported by an external server.



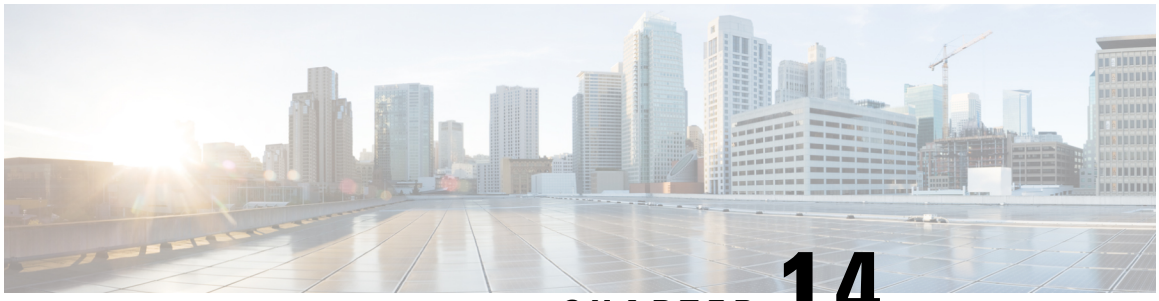
The external server solution has the following characteristics:

- Controller-less environment
- Manual management of security certificates
- Three supported use-cases:
  - Continuous collection of logs from selected switches

- TAC-assisted effort to deploy and upload logs to Cisco servers.
- Limited on-premise processing



Note Contact Cisco TAC for information regarding the setup and collection of log files in an external server.



CHAPTER 14

Configuring SPAN

This chapter contains the following sections:

- Information About SPAN, on page 179
- Guidelines and Limitations for SPAN, on page 179
- SPAN Sources, on page 180
- Characteristics of Source Ports, on page 180
- SPAN Destinations, on page 180
- Characteristics of Destination Ports, on page 181
- SPAN and ERSPAN Filtering, on page 181
- SPAN and ERSPAN Sampling, on page 183
- SPAN and ERSPAN Truncation, on page 183
- Creating or Deleting a SPAN Session, on page 184
- Configuring an Ethernet Destination Port, on page 184
- Configuring Source Ports, on page 186
- Configuring Source Port Channels or VLANs, on page 186
- Configuring the Description of a SPAN Session, on page 187
- Activating a SPAN Session, on page 188
- Suspending a SPAN Session, on page 188
- Configuring a SPAN Filter, on page 189
- Configuring SPAN Sampling, on page 190
- Configuring SPAN Truncation, on page 192
- Displaying SPAN Information, on page 193

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.

Guidelines and Limitations for SPAN

SPAN have the following guideline and limitation:

- You can monitor the same source interfaces (physical port or port-channel) in multiple local SPAN sessions.
- The Cisco Nexus 3500 Series switches do not support access-group command for SPAN sessions.

SPAN Sources

SPAN sources refer to the interfaces from which traffic can be monitored. The Cisco Nexus device supports Ethernet, port channels, and VLANs as SPAN sources. With VLANs, all supported interfaces in the specified VLAN are included as SPAN sources. You can choose the SPAN traffic in the ingress direction, the egress direction, or both directions for Ethernet 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.

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.
- 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.
- Can be in the same or different VLANs.



Note

- The maximum number of source ports per SPAN session is 128 ports.

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.
- The same destination interface cannot be used for multiple SPAN sessions. However, an interface can act as a destination for a SPAN and an ERSPAN session.

SPAN and ERSPAN Filtering

A SPAN or ERSPAN session can be used to monitor all the traffic on all the source interfaces. This volume of traffic can cause packet drops if there are congestions or if the destination bandwidth is not enough to monitor all the traffic.

Cisco NX-OS Release 6.0(2)A4(1) provides the ability to filter out specific SPAN or ERSPAN traffic flows that must be monitored. Filtering is achieved by creating a filter and attaching it to a SPAN or ERSPAN session. Only the packets that match the filter are mirrored.

Filtering can be of the following types:

- MAC-based
- IP-based
- VLAN-based

Guidelines and Limitations for SPAN and ERSPAN Filtering

SPAN and ERSPAN filtering have the following guidelines and limitations:

- Cisco Nexus 3500 Series switches drop the SPAN copies while spanning an interface in the rx direction and another interface in the tx direction when the traffic starts. It happens due to the default SPAN threshold limit being low and it cannot handle the burst traffic for SPAN. Use the CLI command **hardware profile buffer span-threshold <xx>** to increase the SPAN threshold.



Note Increasing the SPAN threshold affects the shared buffer allocation. It allocates the SPAN buffers from the shared buffer pool.

- The span-threshold least value has been updated from 0 to 2. When you set the span-threshold to a lowest value of 2, the SPAN buffer occupied is 528. When you use the negate command **no hardware profile buffer span-threshold 2**, the span-threshold value is 208. The default value is lesser then the least value of span-threshold.
- When a source interface in a SPAN session is operationally down, then that SPAN session will not go operationally down. This behavior does not impact any functionality
- Configuring two SPAN or ERSPAN sessions on the same source interface with only one filter is not supported. If the same source is used in multiple SPAN or ERSPAN sessions, either all the sessions must have different filters or no sessions should have filters.
- SPAN filtering supports only 16 filters. These filters can be a combination of VLAN-based, IP-based, and MAC-based filters.
- When a SPAN session is configured with a multicast router port being the source port, the destination port sees all the multicast traffic even when there is no traffic that is actually being forwarded to the source port. This is due to a current limitation of the multicast/SPAN implementation.
- SPAN filtering is applicable for all the traffic of the switch except the SPAN source interface traffic.
- You can configure only one IP-based, one MAC-based and one VLAN-based filter per SPAN session.
- The number of filters is further restricted by the number of SPAN sessions and the type of source as follows:
  - A maximum of 8 MAC-based, 8 IP-based or 8 VLAN-based filters can be configured.
  - A maximum of 4 IP-based, 4 MAC-based or 4 VLAN-based filters can be attached to all interface-based SPAN sessions.
  - A maximum of 8 IP-based, 8 MAC-based or 8 VLAN-based filters can be attached to all VLAN-based SPAN sessions.
- Filters can be used only in the ingress direction. This is not configurable.
- A SPAN session must be up to for filters to work.
- You cannot configure filters on ERSPAN-dst sessions.
- You cannot configure filters on Warp SPAN sessions.
- The control-packet filter is always applied in the egress direction.
- The control-packet filter is recommended when both, the source and the destination interfaces of the ERSPAN session are PTP enabled.

SPAN and ERSPAN Control-packet Filtering

Cisco NX-OS Release 6.0(2)A8(9) provides the ability to filter out CPU generated packets going out of the SPAN source interface. Control-packet filter is applied in the egress direction, and is therefore effective on source interfaces enabled for Tx mirroring.

SPAN and ERSPAN Sampling

Cisco NX-OS Release 6.0(2)A4(1) supports sampling of source packets for each SPAN or ERSPAN session. Monitoring only a sample number of source packets helps reduce SPAN or ERSPAN bandwidth. This sample is defined by a range that you can configure. For example, if you configure the range as 2, 1 out of every 2 source packets will be spanned. Similarly, if you configure the range as 1023, 1 out of every 1023 packets will be spanned. This method provides an accurate count of SPAN or ERSPAN source packets, but it does not include any time-related information about the spanned packets.

By default, SPAN and ERSPAN sampling are disabled. To use sampling, you must enable it for each SPAN or ERSPAN session.

Guidelines and Limitations for SPAN and ERSPAN Sampling

SPAN and ERSPAN sampling have the following guidelines and limitations:

- Sampling is only supported for local and ERSPAN-src sessions.
- Sampling is not supported for ERSPAN-dst sessions.
- Sampling is not supported for Warp SPAN sessions.
- The supported sampling range is from 2 to 1023.

SPAN and ERSPAN Truncation

Cisco NX-OS Release 6.0(2)A4(1) introduces truncation of source packets for each SPAN or ERSPAN session based on the size of their MTU. Truncation helps reduce SPAN or ERSPAN bandwidth by reducing the size of packets monitored. MTU truncation can be set from 64 bytes to 1518 bytes. Any SPAN or ERSPAN packet that is larger than the configured MTU size is truncated to the given size with a 4-byte offset. For example, if you configure the MTU as 300 bytes, the maximum size of the replicated packet is 304 bytes.

By default, SPAN and ERSPAN truncation are disabled. To use truncation, you must enable it for each SPAN or ERSPAN session.

Guidelines and Limitations for SPAN and ERSPAN Truncation

SPAN and ERSPAN truncation have the following guidelines and limitations:

- Truncation is only supported for local and ERSPAN-src sessions.
- Truncation is not supported for ERSPAN-dst sessions.
- Truncation is not supported for Warp SPAN sessions.
- The supported MTU range is from 64 bytes to 1518 bytes.

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **monitor session** *session-number*

DETAILED STEPS

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface ethernet** *slot*/*port*
3. switch(config-if)# **switchport monitor**
4. switch(config-if)# **exit**
5. switch(config)# **monitor session** *session-number*
6. switch(config-monitor)# **destination interface ethernet** *slot*/*port*

DETAILED STEPS

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. **Note** 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. **Note** 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **monitor session** *session-number*
3. switch(config-monitor) # **source interface** *type slot/port* [**rx** | **tx** | **both**]

DETAILED STEPS

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

```
switch# configure terminal
switch(config)# monitor session 2
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **monitor session** *session-number*
3. switch(config-monitor) # **source** {**interface** {**port-channel** | **san-port-channel**} *channel-number* [**rx** | **tx** | **both**] | **vlan** *vlan-range* | **vsan** *vsan-range* }

DETAILED STEPS

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) # **source** {**interface** {**port-channel** | **san-port-channel**} *channel-number* [**rx** | **tx** | **both**] | **vlan** *vlan-range* | **vsan** *vsan-range* }	Configures port channel, SAN port channel, VLAN, or VSAN sources. For VLAN or VSAN 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)# 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)# 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **monitor session** *session-number*
3. switch(config-monitor) # **description** *description*

DETAILED STEPS

Procedure

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

	Command or Action	Purpose
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **no monitor session** {**all** | *session-number*} **shut**

DETAILED STEPS

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**.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **monitor session** {**all** | *session-number*} **shut**

DETAILED STEPS

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) #
```

Configuring a SPAN Filter

You can configure SPAN filters for local and ERSPAN-source sessions only.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **monitor session** *session-number*
3. switch(config-monitor)# **source** {**interface** {**port-channel**} *channel-number* [**rx** | **tx** | **both**] | **vlan** *vlan-range*}
4. switch(config-monitor)# **filter** {**ip** *source-ip-address source-ip-mask destination-ip-address destination-ip-mask*}
5. switch(config-monitor)# **destination interface ethernet** *slot/port*

DETAILED STEPS

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.

	Command or Action	Purpose
Step 3	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.
Step 4	switch(config-monitor)# **filter** {**ip** *source-ip-address source-ip-mask destination-ip-address destination-ip-mask*}	Creates a SPAN filter.
Step 5	switch(config-monitor)# **destination interface ethernet** *slot/port*	Configures the Ethernet SPAN destination port.

Example

The following example shows how to configure an IP-based SPAN filter for a local session:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# monitor session 1
switch(config-monitor)# source interface Ethernet 1/7 rx
switch(config-monitor)# filter ip 10.1.1.1 255.255.255.255 20.1.1.1 255.255.255.255
switch(config-monitor)# destination interface Ethernet 1/48
switch(config-monitor)# no shut
switch(config-monitor)#
```

The following example shows how to configure a VLAN-based SPAN filter for a local session:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# monitor session 3
switch(config-monitor)# source vlan 200
switch(config-monitor)# destination interface Ethernet 1/4
switch(config-monitor)# no shut
switch(config-monitor)#
```

Configuring SPAN Sampling

You can configure sampling for local and ERSPAN-source sessions only.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **monitor session** *session-number*
3. switch(config-monitor)# **source** {**interface** {**port-channel**} *channel-number* [**rx** | **tx** | **both**] | **vlan** *vlan-range*}
4. switch(config-monitor)# **sampling** *sampling-range*
5. switch(config-monitor)# **destination interface ethernet** *slot/port*

DETAILED STEPS

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)# **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.
Step 4	switch(config-monitor)# **sampling** *sampling-range*	Configures a range for spanning packets. If the range is defined as *n*, every nth packet will be spanned. The sampling range is between 2 and 1023.
Step 5	switch(config-monitor)# **destination interface ethernet** *slot/port*	Configures the Ethernet SPAN destination port.

Example

The following example shows how to configure sampling on a VLAN for a local session:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# monitor session 1
switch(config-monitor)# source vlan 100
switch(config-monitor)# sampling 10
switch(config-monitor)# destination interface ethernet 1/48
switch(config-monitor)# no shut
switch(config-monitor)# show monitor session 1
   session 1
---------------
type              : local
state             : up
sampling          : 10
source intf       :
    rx            : Eth1/3        Eth1/7
    tx            :
    both          :
source VLANs      :
    rx            : 100
destination ports : Eth1/48

Legend: f = forwarding enabled, l = learning enabled
```

The following example shows how to configure sampling on an Ethernet interface for a local session:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# monitor session 3
switch(config-monitor)# source interface ethernet 1/8
switch(config-monitor)# sampling 20
switch(config-monitor)# destination interface ethernet 1/4
switch(config-monitor)# show monitor session 3
```

```
   session 3
---------------
type              : local
state             : down (No operational src/dst)
sampling          : 20
source intf       :
    rx            : Eth1/8
    tx            : Eth1/8
    both          : Eth1/8
source VLANs      :
    rx            : 200
destination ports : Eth1/4

Legend: f = forwarding enabled, l = learning enabled
```

Configuring SPAN Truncation

You can configure truncation for local and ERSPAN-source sessions only.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **monitor session** *session-number*
3. switch(config-monitor) # **source** {**interface** {**port-channel**} *channel-number* [**rx** | **tx** | **both**] | **vlan** *vlan-range*}
4. switch(config-monitor) # **mtu** *size*
5. switch(config-monitor)# **destination interface ethernet** *slot/port*

DETAILED STEPS

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) # **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.
Step 4	switch(config-monitor) # **mtu** *size*	Configures the MTU size for truncation. Any SPAN packet that is larger than the configured MTU size is truncated to the configured size with a 4-byte offset. The MTU truncation size is between 64 bytes and 1518 bytes.
Step 5	switch(config-monitor)# **destination interface ethernet** *slot/port*	Configures the Ethernet SPAN destination port.

Example

The following example shows how to configure MTU truncation for a local session:

```
switch# configure terminal
switch(config)# monitor session 5
switch(config-monitor)# source interface ethernet 1/5 both
switch(config-monitor)# mtu 512
switch(config-monitor)# destination interface Ethernet 1/39
switch(config-monitor)# no shut
switch(config-monitor)# show monitor session 5
   session 5
---------------
type              : local
state             : down (No operational src/dst)
mtu               : 512
source intf       :
    rx            : Eth1/5
    tx            : Eth1/5
    both          : Eth1/5
source VLANs      :
    rx            :
destination ports : Eth1/39

Legend: f = forwarding enabled, l = learning enabled
```

Displaying SPAN Information

SUMMARY STEPS

1. switch# **show monitor** [**session** {**all** | *session-number* | **range** *session-range*} [**brief**]]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **show monitor** [**session** {**all** | *session-number* | **range** *session-range*} [**brief**]]	Displays the SPAN configuration.

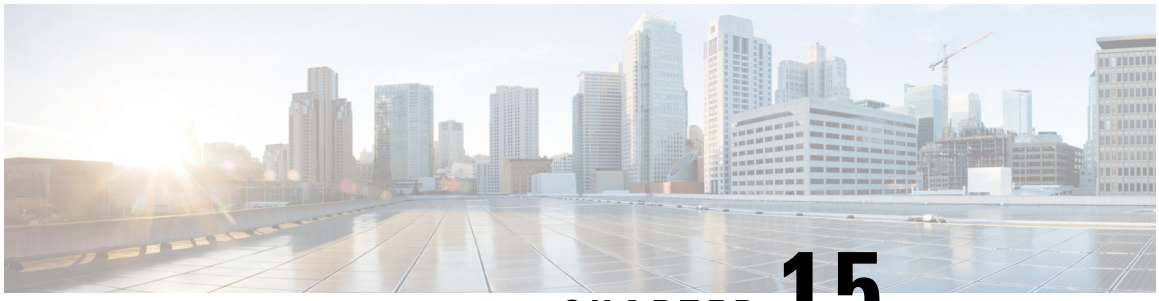
Example

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            :
destination ports : Eth3/1
```



CHAPTER 15

Configuring Warp SPAN

This chapter contains the following sections:

- Information About Warp SPAN, on page 195
- Guidelines and Limitations for Warp Span, on page 196
- Configuring Warp SPAN, on page 197
- Verifying Warp SPAN Mode Configuration, on page 198
- Feature History for Warp SPAN, on page 199

Information About Warp SPAN

Warp SPAN is AlgoBoost feature that spans the traffic coming into a dedicated port to a group of ports at very low latency. In Warp SPAN, traffic arriving at one dedicated ingress port is replicated to a user configurable group of egress ports. The packet replication happens without any filters or lookup mechanisms. Unlike normal or Warp mode traffic forwarding, the incoming traffic is replicated before any traffic classification or ACL processing occurs. Because traffic bypasses these processes, the latency for the replicated packets is as low as 50ns. The Warp SPAN functions independently and simultaneously to normal traffic forwarding. For example, the incoming source traffic can be switched, routed, multicast replicated, and so on, while at the same time this incoming traffic is warp spanned to multiple destination ports.

The original traffic ingressing the dedicated source port is forwarded normally with nominal switch latency, along with the Warp SPAN traffic at about 50ns to the configured destination ports. Warp SPAN can be enabled both in normal traffic forwarding mode and Warp mode.

The source can be monitored only in the ingress direction and is not configurable. The source port is configured automatically as soon as you configure the Warp SPAN session.

You configure the dedicated source Layer 2/Layer 3 port (must be Ethernet port 1/36) with standard configuration as required by the network.

You configure destination ports similar to any regular SPAN destination port. The destination ports cannot be used as regular Layer 2/Layer 3 ports. Destination ports must be configured in groups of four, so you can create a maximum of 12 groups with a total of 47 destination ports (one port—port 1/36—is the fixed source port). See the following table.

Table 29: Warp SPAN Groups

Group	Destination Ports
1	1-4

Group	Destination Ports
2	5-8
3	9-12
4	13-16
5	17-20
6	21-24
7	25-28
8	29-32
9	33-35 [1]
10	37-40
11	41-44
12	45-48

[1] Port 36 is the dedicated source port.

Guidelines and Limitations for Warp Span

Warp SPAN has the following configuration guidelines and limitations:

- Source and destination Warp SPAN ports must all be 10G.
- The source port is not configurable and is fixed as Ethernet port 1/36.
- You can create a maximum of 12 groups with a total of 47 destination ports. All of the groups have four ports, except for group 9, which has only three ports and excludes port 1/36 (the fixed source port).
- All four ports in a group must be configured with the **switchport monitor** command before they can be grouped in a SPAN destination group.
- Warp SPAN does not allow the destination group to be configured unless all of the ports are administratively up. After the group has been configured, you can bring up or down any of the ports in the SPAN destination group. If you copy a working warp configuration that has one or more ports in the administratively down state and paste that configuration back in the configuration file of the same switch, Warp SPAN logs the following error:

  ```
  ERROR: Cannot configure group with member interfaces in admin DOWN state
  ```

- The use of the same source interface on Warp SPAN and ERSPAN is not supported.

Configuring Warp SPAN

You configure Warp SPAN by enabling it and then configuring its destination groups.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config-monitor)# **interface ethernet** *port/slot*
3. switch(config-if)# **switchport monitor**
4. switch(config-if)# **no shutdown**
5. switch(config)# **monitor session warp**
6. switch(config)# **no shutdown**
7. switch(conifig-monitor)# **destination group** *group-number*
8. (Optional) switch(config-if)# **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config-monitor)# **interface ethernet** *port/slot*	Enters interface configuration mode for the specified interface. **Note** You can specify a range to configure multiple interfaces at once.
Step 3	switch(config-if)# **switchport monitor**	Sets the interface to monitor mode. Priority flow control (PFC) is disabled when the port is configured as a SPAN destination.
Step 4	switch(config-if)# **no shutdown**	Brings the interface administratively up.
Step 5	switch(config)# **monitor session warp**	Enables Warp SPAN on the interface.
Step 6	switch(config)# **no shutdown**	Brings the interface administratively up.
Step 7	switch(conifig-monitor)# **destination group** *group-number*	Configures the destination group. **Note** You can create a maximum of 12 groups with a total of 47 destination ports. All of the groups have four ports, except for group 9, which has only three ports and excludes port 1/36 (the fixed source port).
Step 8	(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 configure destination SPAN ports 1/1-4 for Warp SPAN:

```
switch# configure terminal
switch(config-monitor)# interface ethernet 1/1-4
switch(config-if-range)# switchport monitor
switch(config-if-range)# no shutdown
switch(config)# monitor session warp
switch(config)# no shutdown
switch(config-monitor)# destination group 1
switch(config-if-range)# copy running-config startup-config
```

Verifying Warp SPAN Mode Configuration

You can verify the Warp SPAN mode configuration.

SUMMARY STEPS

1. switch(config)# **show monitor session** {*number* | **all** | *range*}
2. switch(config)# **show monitor session warp**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch(config)# **show monitor session** {*number* | **all** | *range*}	Displays information about a specific SPAN session, all SPAN sessions, or a range of SPAN sessions.
Step 2	switch(config)# **show monitor session warp**	Displays information about only the Warp SPAN sessions.

Example

This example shows how to display information about all SPAN sessions and only the Warp SPAN sessions:

```
switch(config)# show monitor session all
session warp
---------------
type : local
state : up
source intf :
rx : Eth1/36
tx :
both :
source VLANs :
rx :
destination ports : Eth1/1 Eth1/2 Eth1/3 Eth1/4
```

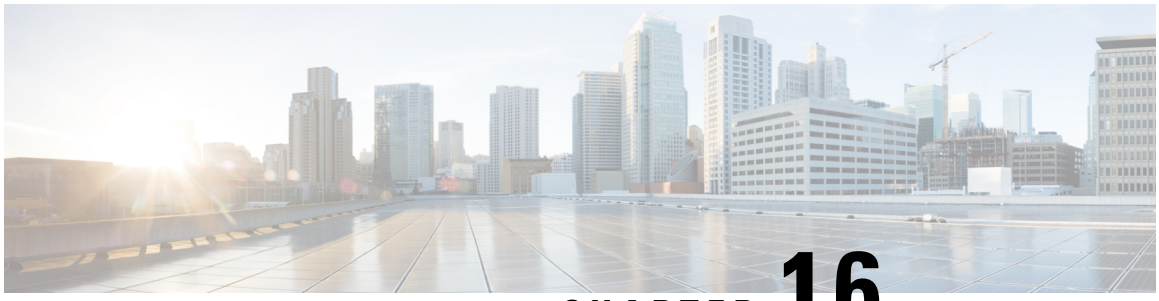
```
Legend: f = forwarding enabled, l = learning enabled


switch(config)# show monitor session warp
session warp
---------------
type : local
state : up
source intf :
rx : Eth1/36
tx :
both :
source VLANs :
rx :
destination ports : Eth1/1 Eth1/2 Eth1/3 Eth1/4

Legend: f = forwarding enabled, l = learning enabled
```

Feature History for Warp SPAN

Feature Name	Release	Feature Information
Warp SPAN	5.0(3)A1(2)	This feature was introduced.



CHAPTER 16

Configuring ERSPAN

This chapter contains the following sections:

- Information About ERSPAN, on page 201
- Prerequisites for ERSPAN, on page 203
- Guidelines and Limitations for ERSPAN, on page 204
- Default Settings for ERSPAN, on page 205
- Configuring ERSPAN, on page 206
- Configuration Examples for ERSPAN, on page 219
- Additional References, on page 220

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.

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.

ERSPAN Types

ERSPAN Type III supports all of the ERSPAN Type II features and functionality and adds these enhancements:

- Provides timestamp information in the ERSPAN Type III header that can be used to calculate packet latency among edge, aggregate, and core switches.
- Identifies possible traffic sources using the ERSPAN Type III header fields.

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:

- Source Ports—A source port is a port monitored for traffic analysis. You can configure source ports in any VLAN, and trunk ports can be configured as source ports and mixed with nontrunk source ports.

- Source VLANs—A source VLAN is a virtual local area network (VLAN) that is monitored for traffic analysis.
- Source VSANs—A source VSAN is a virtual storage area network (VSAN) that is monitored for traffic analysis.

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.

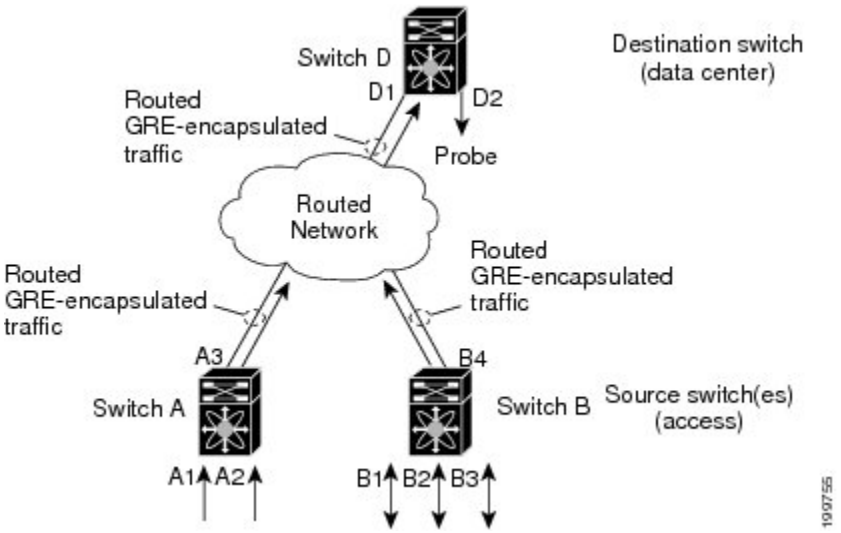
When configuring ERSPAN source sessions, you must configure the destination IP address. When configuring ERSPAN destination sessions, you must configure the source IP address. See ERSPAN Sources, on page 201 for the properties of source sessions and ERSPAN Destinations, on page 202 for the properties of destination sessions.



Note Only eight unidirectional, or four bidirectional ERSPAN or SPAN source sessions can run simultaneously across all switches. Only 20 ERSPAN destination sessions can run simultaneously across all switches.

The following figure shows an ERSPAN configuration.

Figure 2: ERSPAN Configuration



Multiple ERSPAN Sessions

You can define up to eight unidirectional ERSPAN source or SPAN sessions, or four bidirectional ERSPAN source or SPAN sessions at one time. You can shut down any unused ERSPAN sessions.

For information about shutting down ERSPAN sessions, see Shutting Down or Activating an ERSPAN Session, on page 211.

ERSPAN Marker Packet

The type III ERSPAN header carries a hardware generated 32-bit timestamp. This timestamp field wraps periodically. When the switch is set to 1 ns granularity, this field wraps every 4.29 seconds. Such a wrap time makes it difficult to interpret the real value of the timestamp.

To recover the real value of the ERSPAN timestamp, Cisco NX-OS Release 6.0(2)A4(1) introduces a periodical marker packet to carry the original UTC timestamp information and provide a reference for the ERSPAN timestamp. The marker packet is sent out in 1-second intervals. Therefore, the destination site can detect the 32-bit wrap by checking the difference between the timestamp of the reference packet and the packet order.

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:

- ERSPAN supports the following:
  - ERSPAN source session type (Packets are encapsulated as GRE-tunnel packets and sent on the IP network.)
  - ERSPAN destination session type (Support for decapsulating the ERSPAN packet is available. The encapsulated packet is decapsulated at the destination box and the plain decapsulated packet is spanned to a front panel port at the ERSPAN terminating point.)
- ERSPAN source sessions are shared with local SPAN sessions. You can configure a maximum of eight ERSPAN source or SPAN source sessions in a single direction; If both receive and transmit sources are configured in the same session, it counts as two sessions and you can configure four such bidirectional sessions at one time.
- If you install Cisco 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 Cisco NX-OS 5.0(3)U2(2).

  For information about a similar SPAN limitation, see Guidelines and Limitations for SPAN, on page 179.
- ERSPAN is not supported for packets generated by the supervisor.
- ERSPAN sessions are terminated identically at the destination router.
- ERSPAN is not supported for management ports.
- A destination port can be configured in multiple 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.
  - The 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 include:
  - Traffic that results from flooding

    - Broadcast and multicast traffic

- When Nexus 3548 is the ERSPAN destination, GRE headers are stripped off before sending mirrored packets out of the terminating point.
- ERSPAN does not support 1588 granularity mode, and rejects this mode if selected.
- ERSPAN supports 100 microseconds (µs), 100 nanoseconds (ns), and ns granularity.
- ERSPAN sends all timestamps in 32-bit format. Therefore, the timestamp field will wrap periodically. When the switch is set to ns granularity, this field will wrap every 4.29 seconds.
- A Layer 3 subinterface cannot be configured as an ERSPAN source interface.
- All ERSPAN sources terminating in a single destination box must use the same destination IP address.
- You cannot configure different source IP addresses in different ERSPAN destination sessions.
- Layer 3 switched traffic from VLAN X to VLAN Y, which is spanned through the ERSPAN source in either the Rx or Tx direction, will carry VLAN information in the ERSPAN header of VLAN X (the VLAN before Layer 3 switching or ingress VLAN).
- Multicast flood packets that do not go out of the ERSPAN source interface, which is configured for the egress (Tx) direction, can still reach the ERSPAN destination. This is because egress spanned packets are spanned before the original egress port is selectively enabled to receive specific frames and drop others, whereas the span for the Nexus 3548 switch application-specific integrated circuit (ASIC) is based on the monitor port's property. As a result, the spanned packet is still sent to the remote destination. This is expected behavior from platforms specific to multicast flood and is not seen for other traffic streams.
- Replicated multicast packets sent out of the ERSPAN source in the Tx direction are not sent to the ERSPAN destination.
- You can monitor the same source interfaces (physical port or port-channel) in multiple ERSPAN (type 2 or type 3) sessions.
- Configuring IP Filter on ERSPAN or Local SPAN with VLAN as source is not supported.

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.

SUMMARY STEPS

1. **configure terminal**
2. **monitor erspan origin ip-address** *ip-address* **global**
3. **monitor erspan granularity 100_ns**{**100_us**|**100_ns**|**ns**}
4. **no monitor session** {*session-number* | **all**}
5. **monitor session** {*session-number* | **all**} **type erspan-source**
6. **header-type** *version*
7. **description** *description*
8. **source** {[**interface**[*type slot/port*[*-port*][*, type slot/port[-port*]]] [**port-channel** *channel-number*]] | [**vlan** {*number* | *range*}]} [**rx** | **tx** | **both**]
9. (Optional) Repeat Step 6 to configure all ERSPAN sources.
10. **destination ip** *ip-address*
11. **erspan-id** *erspan-id*
12. **vrf** *vrf-name*
13. (Optional) **ip ttl** *ttl-number*
14. (Optional) **ip dscp** *dscp-number*
15. **no shut**
16. (Optional) **show monitor session** {**all** | *session-number* | **range** *session-range*}
17. (Optional) **show running-config monitor**
18. (Optional) **show startup-config monitor**
19. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

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

	Command or Action	Purpose
	`switch# config t` `switch(config)#`	
Step 2	**monitor erspan origin ip-address** *ip-address* **global** **Example:** `switch(config)# monitor erspan origin ip-address 10.0.0.1 global`	Configures the ERSPAN global origin IP address.
Step 3	**monitor erspan granularity 100_ns**{**100_us**|**100_ns**|**ns**} **Example:** `switch(config)# monitor erspan granularity 100_ns`	Configures the granularity of all ERSPAN sessions.
Step 4	**no monitor session** {*session-number* | **all**} **Example:** `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 5	**monitor session** {*session-number* | **all**} **type erspan-source**
	Example:
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		Configures an ERSPAN source session.

Step 6	**header-type** *version* **Example:** `switch(config-erspan-src)# header-type 3`	(Optional) Changes the ERSPAN source session from Type II to Type III.
Step 7	**description** *description* **Example:** `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 8	**source** {[**interface**[*type slot/port*[*-port*][, *type slot/port[-port*]]] [**port-channel** *channel-number*]] | [**vlan** {*number* | *range*}]} [**rx** | **tx** | **both**]
	Example:
	`switch(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	Example:
	`switch(config-erspan-src)# source interface port-channel 2`
	Example:
	`switch(config-erspan-src)# source interface sup-eth 0 both`
	Example:
	`switch(config-monitor)# source interface ethernet 101/1/1-3`	

	Command or Action	Purpose
Step 9	(Optional) Repeat Step 6 to configure all ERSPAN sources.	—
Step 10	**destination ip** *ip-address* **Example:** `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 11	**erspan-id** *erspan-id* **Example:** `switch(config-erspan-src)# erspan-id 5`	Configures the ERSPAN ID for the ERSPAN source session. The ERSPAN range is from 1 to 1023. This ID uniquely identifies a source and destination ERSPAN session pair. The ERSPAN ID configured in the corresponding destination ERSPAN session must be same as the one configured in the source session.
Step 12	**vrf** *vrf-name* **Example:** `switch(config-erspan-src)# vrf default`	Configures the VRF that the ERSPAN source session uses for traffic forwarding.
Step 13	(Optional) **ip ttl** *ttl-number* **Example:** `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 14	(Optional) **ip dscp** *dscp-number* **Example:** `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 15	**no shut** **Example:** `switch(config-erspan-src)# no shut`	Enables the ERSPAN source session. By default, the session is created in the shut state. **Note** Only two ERSPAN source sessions can be running simultaneously.
Step 16	(Optional) **show monitor session** {**all** | *session-number* | **range** *session-range*} **Example:** `switch(config-erspan-src)# show monitor session 3`	Displays the ERSPAN session configuration.
Step 17	(Optional) **show running-config monitor** **Example:** `switch(config-erspan-src)# show running-config monitor`	Displays the running ERSPAN configuration.
Step 18	(Optional) **show startup-config monitor** **Example:** `switch(config-erspan-src)# show startup-config monitor`	Displays the ERSPAN startup configuration.

	Command or Action	Purpose
Step 19	(Optional) **copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring an ERSPAN Destination Session

You can configure an ERSPAN destination session to copy packets from a source IP address to destination ports on the local device. By default, ERSPAN destination sessions are created in the shut state.

Before you begin

Ensure that you have already configured the destination ports in monitor mode.

SUMMARY STEPS

1. **config t**
2. **interface ethernet** *slot/port*[*-port*]
3. **switchport**
4. **switchport mode** [**access** | **trunk**]
5. **switchport monitor**
6. Repeat Steps 2 to 5 to configure monitoring on additional ERSPAN destinations.
7. **no monitor session** {*session-number* | **all**}
8. **monitor session** {*session-number* | **all**} **type erspan-destination**
9. **description** *description*
10. **source ip** *ip-address*
11. **destination** {[**interface** [*type slot/port*[*-port*], [*type slot/port* [*port*]]]}
12. **erspan-id** *erspan-id*
13. **no shut**
14. (Optional) **show monitor session** {**all** | *session-number* | **range** *session-range*}
15. (Optional) **show running-config monitor**
16. (Optional) **show startup-config monitor**
17. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**config t**
	Example:
	`switch# config t`
	`switch(config)#`
		Enters global configuration mode.

	Command or Action	Purpose

Step 2	**interface ethernet** *slot/port*[*-port*]
	Example:
	`switch(config)# interface ethernet 2/5`
	`switch(config-if)#`
		Enters interface configuration mode on the selected slot and port or range of ports.

Step 3	**switchport** **Example:** `switch(config-if)# switchport`	Configures switchport parameters for the selected slot and port or range of ports.
Step 4	**switchport mode** [**access** | **trunk**] **Example:** `switch(config-if)# switchport mode trunk`	Configures the following switchport modes for the selected slot and port or range of ports: • access • trunk
Step 5	**switchport monitor** **Example:** `switch(config-if)# switchport monitor`	Configures the switch interface in monitor mode. To configure an interface to be an ERSPAN or SPAN destination (using the **destination interface ethernet** *interface* command), it must first be configured in monitor mode.
Step 6	Repeat Steps 2 to 5 to configure monitoring on additional ERSPAN destinations.	—
Step 7	**no monitor session** {*session-number* | **all**} **Example:** `switch(config-if)# no monitor session 3`	Clears the configuration of the specified ERSPAN session. The new session configuration is added to the existing session configuration.

Step 8	**monitor session** {*session-number* | **all**} **type erspan-destination**
	Example:
	`switch(config-if)# monitor session 3 type erspan-destination`
	`switch(config-erspan-dst)#`
		Configures an ERSPAN destination session.

Step 9	**description** *description* **Example:** `switch(config-erspan-dst)# description erspan_dst_session_3`	Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters.
Step 10	**source ip** *ip-address* **Example:** `switch(config-erspan-dst)# source ip 10.1.1.1`	Configures the source IP address in the ERSPAN session. Only one source IP address is supported per ERSPAN destination session. This IP address must match the destination IP address that is configured in the corresponding ERSPAN source session.

	Command or Action	Purpose
Step 11	**destination** {[**interface** [*type slot/port*[*-port*], [*type slot/port* [*port*]]]} **Example:** `switch(config-erspan-dst)# destination interface ethernet 2/5`	Configures a destination for copied source packets. You can configure only interfaces as a destination. **Note** You can configure destination ports as trunk ports.
Step 12	**erspan-id** *erspan-id* **Example:** `switch(config-erspan-dst)# erspan-id 5`	Configures the ERSPAN ID for the ERSPAN session. The range is from 1 to 1023. This ID uniquely identifies a source and destination ERSPAN session pair. The ERSPAN ID configured in the corresponding destination ERSPAN session must be same as the one configured in the source session.
Step 13	**no shut** **Example:** `switch(config)# no shut`	Enables the ERSPAN destination session. By default, the session is created in the shut state. **Note** Only 16 active ERSPAN destination sessions can be running simultaneously.
Step 14	(Optional) **show monitor session** {**all** | *session-number* | **range** *session-range*} **Example:** `switch(config)# show monitor session 3`	Displays the ERSPAN session configuration.
Step 15	(Optional) **show running-config monitor** **Example:** `switch(config-erspan-src)# show running-config monitor`	Displays the running ERSPAN configuration.
Step 16	(Optional) **show startup-config monitor** **Example:** `switch(config-erspan-src)# show startup-config monitor`	Displays the ERSPAN startup configuration.
Step 17	(Optional) **copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Shutting Down or Activating an ERSPAN Session

You can shut down ERSPAN sessions to discontinue the copying of packets from sources to destinations. 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.

SUMMARY STEPS

1. **configuration terminal**
2. **monitor session** {*session-range* | **all**} **shut**
3. **no monitor session** {*session-range* | **all**} **shut**
4. **monitor session** *session-number* **type erspan-source**
5. **monitor session** *session-number* **type erspan-destination**
6. **shut**
7. **no shut**
8. (Optional) **show monitor session all**
9. (Optional) **show running-config monitor**
10. (Optional) **show startup-config monitor**
11. (Optional) **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configuration terminal**
	Example:
	`switch# configuration terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**monitor session** {*session-range* | **all**} **shut** **Example:** `switch(config)# monitor session 3 shut`	Shuts down the specified ERSPAN sessions. The session range is from 1 to 48.. By default, sessions are created in the shut state.
Step 3	**no monitor session** {*session-range* | **all**} **shut** **Example:** `switch(config)# no monitor session 3 shut`	Resumes (enables) the specified ERSPAN sessions. The session range is from 1 to 48.. By default, sessions are created in the shut state. . **Note** 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**
	Example:
	`switch(config)# monitor session 3 type erspan-source`
	`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.

	Command or Action	Purpose
Step 5	**monitor session** *session-number* **type erspan-destination** **Example:** `switch(config-erspan-src)# monitor session 3 type erspan-destination`	Enters the monitor configuration mode for the ERSPAN destination type.
Step 6	**shut** **Example:** `switch(config-erspan-src)# shut`	Shuts down the ERSPAN session. By default, the session is created in the shut state.
Step 7	**no shut** **Example:** `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** **Example:** `switch(config-erspan-src)# show monitor session all`	Displays the status of ERSPAN sessions.
Step 9	(Optional) **show running-config monitor** **Example:** `switch(config-erspan-src)# show running-config monitor`	Displays the running ERSPAN configuration.
Step 10	(Optional) **show startup-config monitor** **Example:** `switch(config-erspan-src)# show startup-config monitor`	Displays the ERSPAN startup configuration.
Step 11	(Optional) **copy running-config startup-config** **Example:** `switch(config-erspan-src)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring ERSPAN Filtering

You can configure SPAN filters for local and ERSPAN-source sessions only. SPAN and ERSPAN Filtering, on page 181 provides more information about filters.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **monitor session** {*session-number* | **all**} **type erspan-source**
3. switch(config-erspan-src)# **filter** {**ip** *source-ip-address source-ip-mask destination-ip-address destination-ip-mask*}
4. switch(config-erspan-src)# **erspan-id** *erspan-id*
5. switch(config-erspan-src)# **vrf** *vrf-name*

6. switch(config-erspan-src)# **destination ip** *ip-address*
7. switch(config-erspan-src)# **source** [**interface** [*type slot/port*] | **port-channel** *channel-number*] | [**vlan** *vlan-range*] [**rx** | **tx** | **both**]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **monitor session** {*session-number* | **all**} **type erspan-source**	Configures an ERSPAN source session.
Step 3	switch(config-erspan-src)# **filter** {**ip** *source-ip-address source-ip-mask destination-ip-address destination-ip-mask*}	Creates an ERSPAN filter.
Step 4	switch(config-erspan-src)# **erspan-id** *erspan-id*	Configures the ERSPAN ID for the ERSPAN source session. The ERSPAN range is from 1 to 1023. This ID uniquely identifies a source and destination ERSPAN session pair. The ERSPAN ID configured in the corresponding destination ERSPAN session must be same as the one configured in the source session.
Step 5	switch(config-erspan-src)# **vrf** *vrf-name*	Configures the VRF that the ERSPAN source session uses for traffic forwarding.
Step 6	switch(config-erspan-src)# **destination ip** *ip-address*	Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session.
Step 7	switch(config-erspan-src)# **source** [**interface** [*type slot/port*] | **port-channel** *channel-number*] | [**vlan** *vlan-range*] [**rx** | **tx** | **both**]	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. 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. You can specify the traffic direction to copy as ingress, egress, or both. The default direction is both.

Example

The following example shows how to configure an MAC-based filter for an ERSPAN-source session:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# monitor session 2 type erspan-source
switch(config-erspan-src)# filter abcd.ef12.3456 1111.2222.3333 1234.5678.9012 1111.2222.3333
switch(config-erspan-src)# erspan-id 20
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 200.1.1.1
```

```
switch(config-erspan-src)# source interface Ethernet 1/47 rx
switch(config-erspan-src)# no shut
switch(config-erspan-src)#
```

The following example shows how to configure a VLAN-based filter for an ERSPAN-source session:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# monitor session 2 type erspan-source
switch(config-erspan-src)# filter abcd.ef12.3456 1111.2222.3333 1234.5678.9012 1111.2222.3333
switch(config-erspan-src)# erspan-id 21
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 200.1.1.1
switch(config-erspan-src)# source interface Ethernet 1/47 rx
switch(config-erspan-src)# source vlan 315
switch(config-erspan-src)# mtu 200
switch(config-erspan-src)# no shut
switch(config-erspan-src)#
```

Configuring ERSPAN Sampling

You can configure sampling for local and ERSPAN-source sessions only. SPAN and ERSPAN Sampling, on page 183 provides more information about sampling.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **monitor session** {*session-number* | **all**} **type erspan-source**
3. switch(config-erspan-src)# **sampling** *sampling-range*
4. switch(config-erspan-src)# **erspan-id** *erspan-id*
5. switch(config-erspan-src)# **vrf** *vrf-name*
6. switch(config-erspan-src)# **destination ip** *ip-address*
7. switch(config-erspan-src)# **source** [**interface** *type slot/port* | **port-channel** *channel-number*] | [**vlan** *vlan-range*] [**rx** | **tx** | **both**]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **monitor session** {*session-number* | **all**} **type erspan-source**	Configures an ERSPAN source session.
Step 3	switch(config-erspan-src)# **sampling** *sampling-range*	Configures a range for spanning packets. If the range is defined as n, every nth packet will be spanned. The sampling range is between 2 and 1023.
Step 4	switch(config-erspan-src)# **erspan-id** *erspan-id*	Configures the ERSPAN ID for the ERSPAN source session. The ERSPAN range is from 1 to 1023. This ID uniquely identifies a source and destination ERSPAN

	Command or Action	Purpose
		session pair. The ERSPAN ID configured in the corresponding destination ERSPAN session must be same as the one configured in the source session.
Step 5	switch(config-erspan-src)# **vrf** *vrf-name*	Configures the VRF that the ERSPAN source session uses for traffic forwarding.
Step 6	switch(config-erspan-src)# **destination ip** *ip-address*	Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session.
Step 7	switch(config-erspan-src)# **source** [**interface** *type slot/port* | **port-channel** *channel-number*] | [**vlan** *vlan-range*] [**rx** | **tx** | **both**]	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. 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. You can specify the traffic direction to copy as ingress, egress, or both. The default direction is both.

Example

The following example shows how to configure sampling for an ERSPAN-source session:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# monitor session 2 type erspan-source
switch(config-erspan-src)# sampling 40
switch(config-erspan-src)# erspan-id 30
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 200.1.1.1
switch(config-erspan-src)# source interface ethernet 1/47
switch(config-erspan-src)# show monitor session 2
session 2
---------------
type : erspan-source
state : up
granularity : 100 microseconds
erspan-id : 30
vrf-name : default
destination-ip : 200.1.1.1
ip-ttl : 255
ip-dscp : 0
header-type : 2
mtu : 200
sampling : 40
origin-ip : 150.1.1.1 (global)
source intf :
rx : Eth1/47
tx : Eth1/47
both : Eth1/47
source VLANs :
rx : 315
switch(config-erspan-src)#
```

Configuring ERSPAN Truncation

You can configure truncation for local and ERSPAN-source sessions only. SPAN and ERSPAN Truncation, on page 183 provides more information about truncation.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **monitor session** {*session-number* | **all**} **type erspan-source**
3. switch(config-erspan-src)# **mtu** *size*
4. switch(config-erspan-src)# **erspan-id** *erspan-id*
5. switch(config-erspan-src)# **vrf** *vrf-name*
6. switch(config-erspan-src)# **destination ip** *ip-address*
7. switch(config-erspan-src)# **source** [**interface** *type slot/port* | **port-channel** *channel-number*] | [**vlan** *vlan-range*] [**rx** | **tx** | **both**]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **monitor session** {*session-number* | **all**} **type erspan-source**	Configures an ERSPAN source session.
Step 3	switch(config-erspan-src)# **mtu** *size*	Configures the MTU size for truncation. Any SPAN packet that is larger than the configured MTU size is truncated to the configured size with a 4-byte offset. The MTU truncation size is between 64 bytes and 1518 bytes.
Step 4	switch(config-erspan-src)# **erspan-id** *erspan-id*	Configures the ERSPAN ID for the ERSPAN source session. The ERSPAN range is from 1 to 1023. This ID uniquely identifies a source and destination ERSPAN session pair. The ERSPAN ID configured in the corresponding destination ERSPAN session must be same as the one configured in the source session.
Step 5	switch(config-erspan-src)# **vrf** *vrf-name*	Configures the VRF that the ERSPAN source session uses for traffic forwarding.
Step 6	switch(config-erspan-src)# **destination ip** *ip-address*	Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session.
Step 7	switch(config-erspan-src)# **source** [**interface** *type slot/port* | **port-channel** *channel-number*] | [**vlan** *vlan-range*] [**rx** | **tx** | **both**]	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.

	Command or Action	Purpose
		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. You can specify the traffic direction to copy as ingress, egress, or both. The default direction is both.

Example

The following example shows how to configure MTU truncation for an ERSPAN-source session:

```
switch# configure terminal
switch(config)# monitor session 6 type erspan-source
switch(config-erspan-src)# mtu 1096
switch(config-erspan-src)# erspan-id 40
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 200.1.1.1
switch(config-erspan-src)# source interface ethernet 1/40
switch(config-erspan-src)# show monitor session 6
session 6
---------------
type : erspan-source
state : down (Session admin shut)
granularity : 100 microseconds
erspan-id : 40
vrf-name : default
destination-ip : 200.1.1.1
ip-ttl : 255
ip-dscp : 0
header-type : 2
mtu : 1096
origin-ip : 150.1.1.1 (global)
source intf :
rx : Eth1/40
tx : Eth1/40
both : Eth1/40
source VLANs :
rx :
```

Configuring an ERSPAN Marker Packet

Use the following commands to configure an ERSPAN marker packet:

Command	Purpose
marker-packet seconds	Enables the ERSPAN marker packet for a session. The interval can range from 1 second to 4 seconds.
marker-packet milliseconds	Enables the ERSPAN marker packet for a session. The interval can range from 100 milliseconds to 900 milliseconds, with increments in multiples of 100.
no marker-packet	Disables the ERSPAN marker packet for a session.

Example

This example shows how to enable the ERSPAN marker packet with an interval of 2 seconds:



Note Configuring the interval parameter is optional. If you enable the marker-packet without specifying a parameter, it uses the default or existing interval as the interval value. The **marker-packet** command only enables the marker-packet.

```
switch# configure terminal
switch(config)# monitor erspan origin ip-address 172.28.15.250 global
switch(config)# monitor session 1 type erspan-source
switch(config)# header-type 3
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 9.1.1.2
switch(config-erspan-src)# source interface e1/15 both
switch(config-erspan-src)# marker-packet 2
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
```

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 erspan granularity 100_ns
switch(config-erspan-src)# header-type 3
switch(config)# monitor session 1 type erspan-source
```

```
switch(config-erspan-src)# source interface e14/30
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# destination ip 9.1.1.2
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
switch(config)# show monitor session 1
```



Note **`switch(config)# monitor erspan granularity 100_ns`** and **`switch(config-erspan-src)# header-type 3`** are used only while configuring Type III source sessions.

Configuration Example for an ERSPAN Destination Session

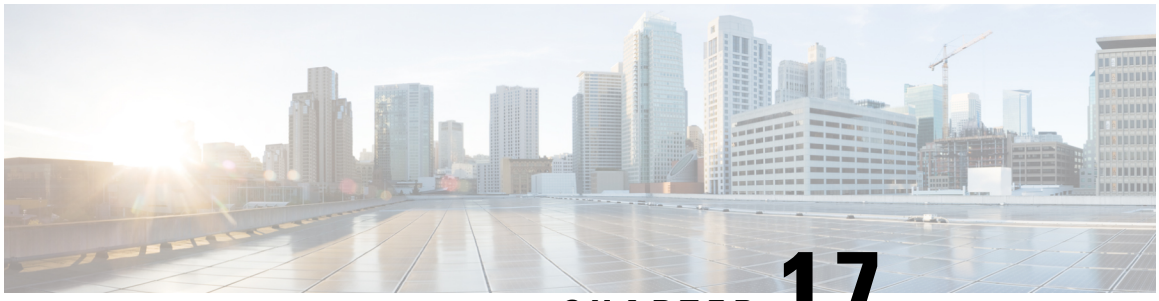
The following example shows how to configure an ERSPAN destination session:

```
switch# config t
switch(config)# interface e14/29
switch(config-if)# no shut
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 2 type erspan-destination
switch(config-erspan-dst)# source ip 9.1.1.2
switch(config-erspan-dst)# destination interface e14/29
switch(config-erspan-src)# erspan-id 1
switch(config-erspan-dst)# no shut
switch(config-erspan-dst)# exit
switch(config)# show monitor session 2
```

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 17

Configuring DNS

This chapter contains the following sections:

- Information About DNS Client , on page 221
- Prerequisites for DNS Clients, on page 222
- Default Settings for DNS Clients, on page 222
- Configuring DNS Clients, on page 222

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 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.

SUMMARY STEPS

1. switch# **configuration terminal**
2. switch(config)# vrf context managment
3. switch(config)# **ip host** *name address1 [address2... address6]*
4. (Optional) switch(config)# **ip domain name** *name* [**use-vrf** *vrf-name*]
5. (Optional) switch(config)# **ip domain-list** *name* [**use-vrf** *vrf-name*]

6. (Optional) switch(config)# **ip name-server** *server-address1 [server-address2... server-address6]* **[use-vrf** *vrf-name*]
7. (Optional) switch(config)# **ip domain-lookup**
8. (Optional) switch(config)# **show hosts**
9. switch(config)# **exit**
10. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

Procedure

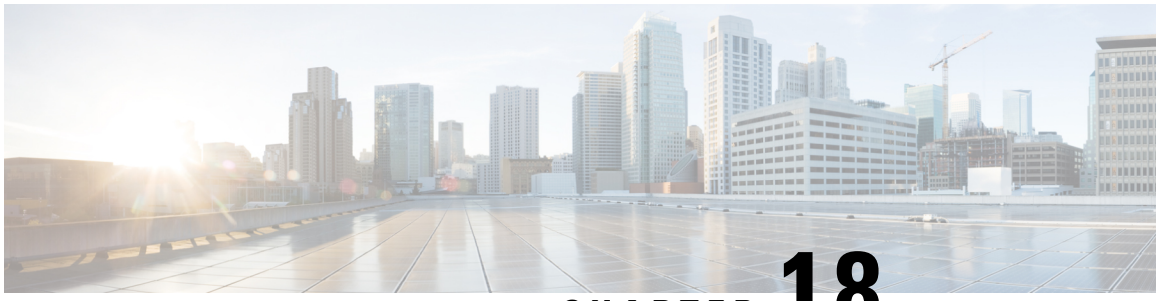
	Command or Action	Purpose
Step 1	switch# **configuration terminal**	Enters global configuration mode.
Step 2	switch(config)# vrf context managment	Specifies a configurable virtual and routing (VRF) name.
Step 3	switch(config)# **ip host** *name address1 [address2... address6]*	Defines up to six static 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 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. 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. 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** *server-address1 [server-address2... server-address6]* **[use-vrf** *vrf-name*]	Defines up to six name servers. The address can be either an IPv4 address or an IPv6 address. 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.

	Command or Action	Purpose
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 18

Configuring Traffic Forwarding Modes

This chapter contains the following sections:

- Information About Warp Mode, on page 225
- Guidelines and Limitations for Warp Mode, on page 225
- Enabling and Disabling Warp Mode, on page 226
- Verifying Warp Mode Status, on page 227
- Feature History for Warp Mode, on page 227

Information About Warp Mode

The Cisco Nexus device uses a hardware component called the Algorithm Boost Engine (Algo Boost Engine) to support a forwarding mechanism, called warp mode. In warp mode, the access path is shortened by consolidating the forwarding table into single table, resulting in faster processing of frames and packets. In warp mode, latency is reduced by up to 20 percent. For more information about the Algo Boost Engine, see Active Buffer Monitoring Overview, on page 229.

Guidelines and Limitations for Warp Mode

Warp mode has the following configuration guidelines and limitations:

- Warp mode provides up to 20 percent better switch latency than normal forwarding.
- In warp mode, unicast route tables are reduced. The route table is reduced from 24000 to 4000 entries. The host table and MAC table are reduced from 64000 to 8000 entries. (The multicast route table remains the same at 8000 entries.)
- In warp mode, the following features are not supported:
  - Egress Routed Access Control Lists (RACLs)
  - Port Access Control Lists (PACLs)
  - Equal-cost Multipathing (ECMP)
  - IP Redirect

Enabling and Disabling Warp Mode

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **hardware profile forwarding-mode warp**
3. (Optional) switch(config)# **copy running-config startup-config**
4. Reload the switch.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **hardware profile forwarding-mode warp**	Enables warp mode on the device. To disable warp mode, use the **no** form of this command. The default is warp mode disabled.
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.
Step 4	Reload the switch.	—

Example

This example shows how to enable warp mode on the device:

```
switch# configuration terminal
switch(config)# hardware profile forwarding-mode warp
Warning: This command will take effect only after saving the configuration (copy r s)
switch(config)# copy running-config startup-config
switch(config)#
```

This example shows how to disable warp mode on the device:

```
switch# configuration terminal
switch(config)# no hardware profile forwarding-mode warp
Warning: This command will take effect only after saving the configuration (copy r s)
switch(config)# copy running-config startup-config
```

Verifying Warp Mode Status

SUMMARY STEPS

1. switch# **show hardware profile forwarding-mode**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **show hardware profile forwarding-mode**	Displays information about warp mode and the host, unicast, multicast, and Layer 2 ternary content addressable memory (TCAM) sizes.

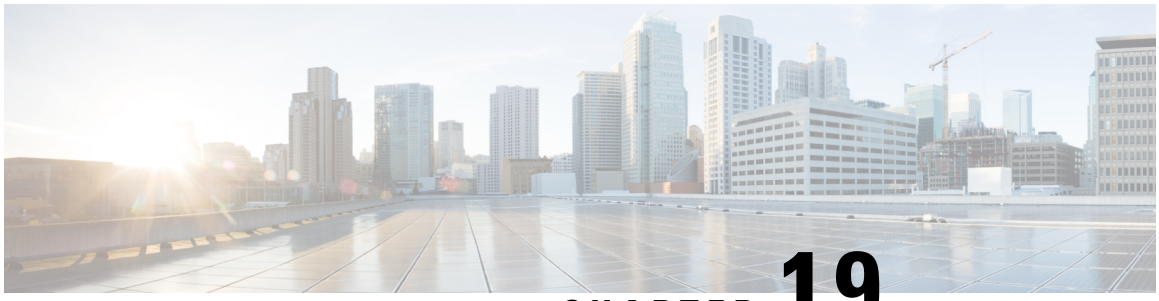
Example

This example shows how to display information about warp mode:

```
switch# show hardware profile forwarding-mode
==========================
 forwarding-mode : warp
==========================
 host      size  = 8192
 unicast   size  = 4096
 multicast size  = 8192
 l2        size  = 8192
switch#
```

Feature History for Warp Mode

Feature Name	Release	Feature Information
Warp Mode	5.0(3)A1(1)	This feature was introduced.



CHAPTER 19

Configuring Active Buffer Monitoring

This chapter contains the following sections:

- Information About Active Buffer Monitoring, on page 229
- Configuring Active Buffer Monitoring, on page 230
- Displaying Buffer Histogram Data, on page 231

Information About Active Buffer Monitoring

Active Buffer Monitoring Overview

The Active Buffer Monitoring feature provides detailed buffer occupancy data to help you detect network congestion, review past events to understand when and how network congestion is affecting network operations, understand historical trending, and identify patterns of application traffic flow.

A hardware component, called the Algorithm Boost Engine (Algo Boost Engine) supports buffer histogram counters for unicast buffer usage per individual port, total buffer usage per buffer block, and multicast buffer usage per buffer block. Each histogram counter has 18 buckets that span across the memory block. The Algo Boost Engine polls buffer usage data every hardware sampling interval (the default is every 4 milliseconds, but you can configure it to be as low as 10 nanoseconds). Based on the buffer utilization, the corresponding histogram counter is incremented. For example, if Ethernet port 1/4 is consuming 500 KB of the buffer, the bucket 2 counter (which represents 384 KB to 768 KB) for Ethernet 1/4 is incremented.

To avoid a counter overflow, the Cisco NX-OS software collects the histogram data every polling interval and maintains it in the system memory. The software maintains the histogram data in the system memory for the last 60 minutes with 1-second granularity. Every hour, the software copies the buffer histogram data from the system memory to the bootflash as a backup.

The Active Buffer Monitoring feature has two modes of operation:

- Unicast mode—The Algo Boost Engine monitors and maintains a buffer histogram for total buffer utilization per buffer block and unicast buffer utilization for all 48 ports.
- Multicast mode—The Algo Boost Engine monitors and maintains buffer histogram data for total buffer utilization per buffer block and multicast buffer utilization per buffer block.

Buffer Histogram Data Access and Collection

After active buffer monitoring is enabled, the device maintains 70 minutes of data—the first 60 minutes (0 to 60 minutes) in the log and another 60 minutes (10 to 70 minutes) in memory.

You can access buffer histogram data using several methods:

- You can access it from the system memory using **show** commands.
- You can integrate the Active Buffer Monitoring feature with Cisco NX-OS Python scripting to collect historical data by copying the data to a server regularly.
- You can access the buffer histogram data using an XML interface.
- You can configure Cisco NX-OS to log a message in the syslog whenever the buffer occupancy exceeds the configured threshold.

Configuring Active Buffer Monitoring

Note If you use NX-API over the front panel port, you must increase the CoPP policy (for HTTP) to allow 3000 PPS traffic. Doing so prevents packet drops, and the CLIs, creating larger outputs, return within the expected time.

Note Active Buffer Monitoring (ABM) is enabled on all front ports, but only default class traffic can be monitored.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **hardware profile buffer monitor** {**unicast** | **multicast**}
3. switch(config)# **hardware profile buffer monitor** {**unicast** | **multicast**} **threshold** *threshold-value*
4. switch(config)# **hardware profile buffer monitor** {**unicast** | **multicast**} **sampling** *sampling-value*
5. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **hardware profile buffer monitor** {**unicast** | **multicast**}	Enables the hardware profile buffer for either unicast or multicast traffic.
Step 3	switch(config)# **hardware profile buffer monitor** {**unicast** | **multicast**} **threshold** *threshold-value*	Generates a syslog entry when the specified maximum buffer size is exceeded. The range is 384–6144 kilobytes

	Command or Action	Purpose
		with 384-kilobyte increments. The default is 90 percent of the total available shared buffer.
Step 4	switch(config)# **hardware profile buffer monitor** {**unicast** | **multicast**} **sampling** *sampling-value*	Specifies to sample data at the specified interval. Range is 10–20,000,000 nanoseconds. The default sampling value is 4 milliseconds.
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 Active Buffer Monitoring for unicast traffic. A threshold value of 384 kilobytes and a sampling value of 5000 nanoseconds is used:

```
switch# configure terminal
switch(config)# hardware profile buffer monitor unicast
switch(config)# hardware profile buffer monitor unicast threshold 384
switch(config)# hardware profile buffer monitor unicast sampling 5000
switch(config)# copy running-config startup-config
```

The following example shows how to configure Active Buffer Monitoring for multicast traffic. A threshold value of 384 kilobytes and a sampling value of 5000 nanoseconds is used.

```
switch# configure terminal
switch(config)# hardware profile buffer monitor multicast
switch(config)# hardware profile buffer monitor multicast threshold 384
switch(config)# hardware profile buffer monitor multicast sampling 5000
switch(config)# copy running-config startup-config
```

Displaying Buffer Histogram Data

SUMMARY STEPS

1. switch# **show hardware profile buffer monitor** [**interface ethernet** *slot/port*] {**brief** | **buffer-block** | **detail** | **multicast** | **summary**}
2. (Optional) switch# **clear hardware profile buffer monitor**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **show hardware profile buffer monitor** [**interface ethernet** *slot/port*] {**brief** | **buffer-block** | **detail** | **multicast** | **summary**}	Displays data collected about the buffer. The keywords are defined as follows: • **brief**—Specifies to show limited information about each interface.

	Command or Action	Purpose
		• **buffer-block**—Specifies to display information about a specific buffer block. • **detail**—Specifies to display all information gathered for each interface. • **interface**—(Optional) Specifies to display information about a specific port. • **multicast**—Specifies to show buffer data for multicast traffic only. • **summary**—Specifies to display summary information about each buffer block. **Note** The show command option **interface** is only valid in unicast mode and the **multicast** option is only valid in multicast mode.
Step 2	(Optional) switch# **clear hardware profile buffer monitor**	Clears the collected buffer data.

Example

This example shows how to display summary information for each buffer block and for all of the buffers combined:

```
switch# show hardware profile buffer monitor summary
Summary CLI issued at: 09/18/2012 07:38:39

                    Maximum buffer utilization detected
                  1sec     5sec    60sec     5min      1hr
                 ------   ------   ------   ------   ------
Buffer Block 1       0KB      0KB      0KB      0KB      N/A

Total Shared Buffer Available = 5049 Kbytes
Class Threshold Limit = 4845 Kbytes
============================================================
Buffer Block 2       0KB      0KB      0KB      0KB      N/A

Total Shared Buffer Available = 5799 Kbytes
Class Threshold Limit = 5598 Kbytes
============================================================
Buffer Block 3       0KB      0KB   5376KB   5376KB      N/A

Total Shared Buffer Available = 5799 Kbytes
Class Threshold Limit = 5598 Kbytes
```

This example shows how to display the maximum buffer utilization of each buffer block and each interface for unicast mode:

```
switch# show hardware profile buffer monitor brief
Brief CLI issued at: 09/18/2012 07:38:29
```

```
                    Maximum buffer utilization detected
                  1sec     5sec    60sec     5min      1hr
                 ------   ------   ------   ------   ------
Buffer Block 1       0KB      0KB      0KB      0KB      N/A

Total Shared Buffer Avaliable = 5049 Kbytes
Class Threshold Limit = 4845 Kbytes
---------------------------------------------------------------
Ethernet1/45         0KB      0KB      0KB      0KB      N/A
Ethernet1/46         0KB      0KB      0KB      0KB      N/A
Ethernet1/47         0KB      0KB      0KB      0KB      N/A
Ethernet1/48         0KB      0KB      0KB      0KB      N/A
Ethernet1/21         0KB      0KB      0KB      0KB      N/A
Ethernet1/22         0KB      0KB      0KB      0KB      N/A
Ethernet1/23         0KB      0KB      0KB      0KB      N/A
Ethernet1/24         0KB      0KB      0KB      0KB      N/A
Ethernet1/9          0KB      0KB      0KB      0KB      N/A
Ethernet1/10         0KB      0KB      0KB      0KB      N/A
Ethernet1/11         0KB      0KB      0KB      0KB      N/A
Ethernet1/12         0KB      0KB      0KB      0KB      N/A
Ethernet1/33         0KB      0KB      0KB      0KB      N/A
Ethernet1/34         0KB      0KB      0KB      0KB      N/A
Ethernet1/35         0KB      0KB      0KB      0KB      N/A
Ethernet1/36         0KB      0KB      0KB      0KB      N/A
===============================================================
Buffer Block 2       0KB      0KB      0KB      0KB      N/A

Total Shared Buffer Avaliable = 5799 Kbytes
Class Threshold Limit = 5598 Kbytes
---------------------------------------------------------------
Ethernet1/17         0KB      0KB      0KB      0KB      N/A
Ethernet1/18         0KB      0KB      0KB      0KB      N/A
Ethernet1/19         0KB      0KB      0KB      0KB      N/A
Ethernet1/20         0KB      0KB      0KB      0KB      N/A
Ethernet1/5          0KB      0KB      0KB      0KB      N/A
Ethernet1/6          0KB      0KB      0KB      0KB      N/A
Ethernet1/7          0KB      0KB      0KB      0KB      N/A
Ethernet1/8          0KB      0KB      0KB      0KB      N/A
Ethernet1/41         0KB      0KB      0KB      0KB      N/A
Ethernet1/42         0KB      0KB      0KB      0KB      N/A
Ethernet1/43         0KB      0KB      0KB      0KB      N/A
Ethernet1/44         0KB      0KB      0KB      0KB      N/A
Ethernet1/29         0KB      0KB      0KB      0KB      N/A
Ethernet1/30         0KB      0KB      0KB      0KB      N/A
Ethernet1/31         0KB      0KB      0KB      0KB      N/A
Ethernet1/32         0KB      0KB      0KB      0KB      N/A
===============================================================
Buffer Block 3       0KB      0KB   5376KB   5376KB      N/A

Total Shared Buffer Avaliable = 5799 Kbytes
Class Threshold Limit = 5598 Kbytes
---------------------------------------------------------------
Ethernet1/13         0KB      0KB      0KB      0KB      N/A
Ethernet1/14         0KB      0KB      0KB      0KB      N/A
Ethernet1/15         0KB      0KB      0KB      0KB      N/A
Ethernet1/16         0KB      0KB      0KB      0KB      N/A
Ethernet1/37         0KB      0KB      0KB      0KB      N/A
Ethernet1/38         0KB      0KB      0KB      0KB      N/A
Ethernet1/39         0KB      0KB      0KB      0KB      N/A
Ethernet1/40         0KB      0KB      0KB      0KB      N/A
Ethernet1/25         0KB      0KB      0KB      0KB      N/A
Ethernet1/26         0KB      0KB      0KB      0KB      N/A
Ethernet1/27         0KB      0KB      0KB      0KB      N/A
```

```
Ethernet1/28          0KB       0KB       0KB       0KB       N/A
Ethernet1/1           0KB       0KB       0KB       0KB       N/A
Ethernet1/2           0KB       0KB       0KB       0KB       N/A
Ethernet1/3           0KB       0KB       0KB       0KB       N/A
Ethernet1/4           0KB       0KB    5376KB    5376KB       N/A
```

This example shows how to display the maximum buffer utilization information of each buffer block for multicast mode:

```
switch# show hardware profile buffer monitor brief
Brief CLI issued at: 09/18/2012 08:30:08

                    Maximum buffer utilization detected
                  1sec      5sec     60sec      5min       1hr
                 ------    ------   ------    ------    ------
Buffer Block 1        0KB       0KB       0KB       0KB       0KB

Total Shared Buffer Avaliable = 5049 Kbytes
Class Threshold Limit = 4845 Kbytes
Mcast Usage 1         0KB       0KB       0KB       0KB       0KB
============================================================
Buffer Block 2        0KB       0KB       0KB       0KB       0KB

Total Shared Buffer Avaliable = 5799 Kbytes
Class Threshold Limit = 5598 Kbytes
Mcast Usage 2         0KB       0KB       0KB       0KB       0KB
============================================================
Buffer Block 3        0KB       0KB       0KB       0KB       0KB

Total Shared Buffer Avaliable = 5799 Kbytes
Class Threshold Limit = 5598 Kbytes
Mcast Usage 3         0KB       0KB       0KB       0KB       0KB
```

The following example shows how to display detailed buffer utilization information of buffer block 3 for multicast mode:

```
switch# show hardware profile buffer monitor multicast 3 detail
Detail CLI issued at: 09/18/2012 08:30:12

Legend -
384KB  - between   1 and 384KB of shared buffer consumed by port
768KB  - between 385 and 768KB of shared buffer consumed by port
307us  - estimated max time to drain the buffer at 10Gbps

Active Buffer Monitoring for Mcast Usage 3 is: Active
KBytes                384  768 1152 1536 1920 2304 2688 3072 3456 3840 4224 4608 4992 5376
 5760 6144
us @ 10Gbps           307  614  921 1228 1535 1842 2149 2456 2763 3070 3377 3684 3991 4298
 4605 4912
                     ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
 ---- ----
09/18/2012 08:30:12     0    0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 08:30:11     0    0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 08:30:10     0    0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 08:30:09     0    0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 08:30:08     0    0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
```

```
09/18/2012 08:30:07      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 08:30:06      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 08:30:05      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 08:30:04      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 08:30:03      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
```

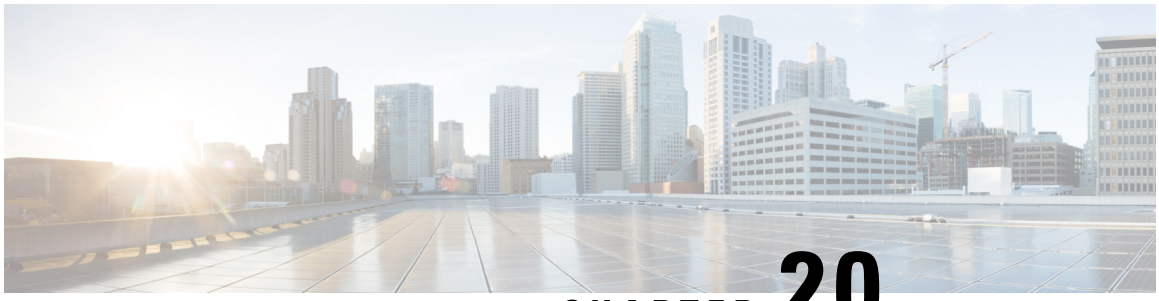
The following example shows how to display detailed buffer data about Ethernet interface 1/4:

```
switch# show hardware profile buffer monitor interface ethernet 1/4 detail
Detail CLI issued at: 09/18/2012 07:38:43

Legend -
384KB  - between   1 and 384KB of shared buffer consumed by port
768KB  - between 385 and 768KB of shared buffer consumed by port
307us  - estimated max time to drain the buffer at 10Gbps

Active Buffer Monitoring for port Ethernet1/4 is: Active
KBytes                384  768 1152 1536 1920 2304 2688 3072 3456 3840 4224 4608 4992 5376
 5760 6144
us @ 10Gbps           307  614  921 1228 1535 1842 2149 2456 2763 3070 3377 3684 3991 4298
 4605 4912
                     ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
 ---- ----
09/18/2012 07:38:42      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:41      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:40      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:39      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:38      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:37      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:36      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:35      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:34      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:33      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:32      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:31      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:30      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:29      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:28      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:27      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:26      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:25      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:24      0    0    0    0    0    0    0    0    0    0    0    0    0
```

```
0    0    0
09/18/2012 07:38:23      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:22      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:21      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:20    177   36    0    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:19      0  143  107    0    0    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:18      0    0   72  178    3    0    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:17      0    0    0    0  176   74    0    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:16      0    0    0    0    0  105  145    0    0    0    0    0    0
0    0    0
09/18/2012 07:38:15      0    0    0    0    0    0   33  179   38    0    0    0    0
0    0    0
09/18/2012 07:38:14      0    0    0    0    0    0    0    0  140  113    0    0    0
0    0    0
09/18/2012 07:38:13      0    0    0    0    0    0    0    0    0   66  178    6    0
0    0    0
09/18/2012 07:38:12      0    0    0    0    0    0    0    0    0    0    0  173   77
0    0    0
09/18/2012 07:38:11      1    0    0    1    0    0    1    0    0    1    0    0  102
42    0    0
09/18/2012 07:38:10      0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0
```



CHAPTER 20

Performing Software Maintenance Upgrades (SMUs)

This chapter includes the following sections:

- About SMUs, on page 237
- Package Management, on page 238
- Prerequisites for SMUs, on page 238
- Guidelines and Limitations for SMUs, on page 239
- Performing a Software Maintenance Upgrade for Cisco NX-OS, on page 239
- Preparing for Package Installation, on page 239
- Copying the Package File to a Local Storage Device or Network Server, on page 241
- Adding and Activating Packages, on page 241
- Committing the Active Package Set, on page 243
- Deactivating and Removing Packages, on page 243
- Displaying Installation Log Information, on page 244

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 3500 Series NX-OS Software Upgrade and Downgrade Guide*.

Note Activating an SMU does not cause any earlier SMUs, or the package to which the SMU applies, to be automatically deactivated.

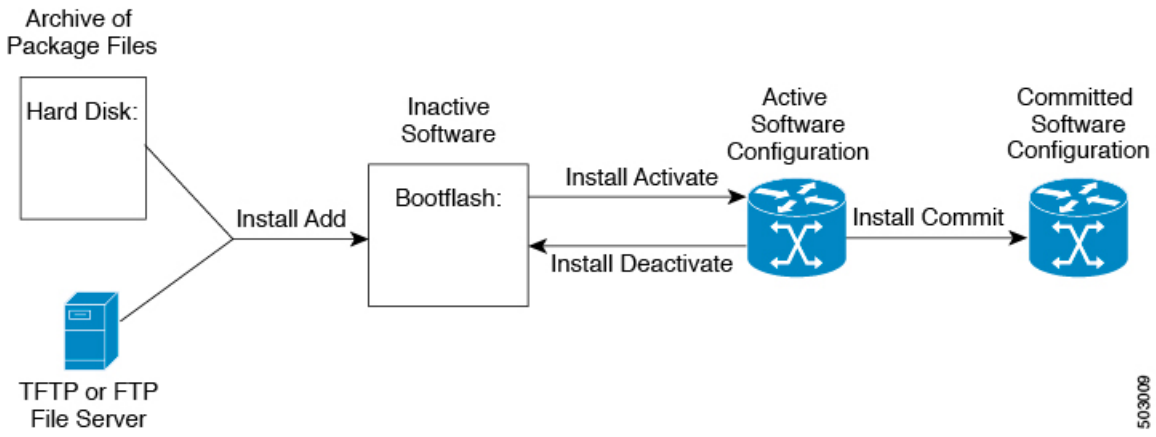
Package Management

The general procedure for adding and activating SMU packages on the device is as follows:

1. Copy the package file or files to a local storage device or file server.
2. Add the package or packages on the device using the **install add** command.
3. Activate the package or packages on the device using the **install activate** command.
4. Commit the current set of packages using the **install commit** command.
5. (Optional) Deactivate and remove the package, when desired.

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 3500 software release, the new image will overwrite both the previous Cisco Nexus 3500 release and the SMU package file.
- Beginning with Cisco NX-OS Release 10.5(1)F, the following guidelines are applicable to SMU:
  - If you need to activate an SMU that is valid and compatible with the image, and the activation fails, then the switch reloads automatically. However, if the SMU is not activated after four tries, do not activate the SMU. Meanwhile, when the switch is ready, a syslog message is displayed indicating that the SMU failed to activate.
  - If you try to install a PID-specific SMU on a PID that it is not meant for, then the following message is displayed: **Install operation failed because SMU is not compatible for this switch model**.
  - If you perform ISSU on a switch along with a tar ball of SMUs containing supported and non-supported SMUs, only the supported SMUs will be installed after ISSU.

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.

SUMMARY STEPS

1. **show install active**
2. **show module**
3. **show clock**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**show install active** **Example:** `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.
Step 2	**show module** **Example:** `switch# show module`	Confirms that all modules are in the stable state.
Step 3	**show clock** **Example:** `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.

SUMMARY STEPS

1. **install add** *filename* [**activate**]
2. (Optional) **show install inactive**
3. **install activate** *filename* [**test**]
4. Repeat Step 3 until all packages are activated.
5. (Optional) **show install active**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**install add** *filename* [**activate**] **Example:** `switch# install add bootflash: n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin`	Unpacks the package software files from the local storage device or network server and adds them to the bootflash: and all active and standby supervisors installed on the device. The *filename* argument can take any of these formats: • **bootflash:***filename* • **tftp://***hostname-or-ipaddress*/*directory-path*/*filename* • **ftp://***username***:***password***@***hostname-or-ipaddress*/*directory-path*/*filename* • **sftp://***hostname-or-ipaddress*/*directory-path*/*filename*
Step 2	(Optional) **show install inactive** **Example:** `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**]
	Example:
	`switch# install activate n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin`
	Example:
	`switch# install activate n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin Install operation 1 completed successfully at Thu Jan 9 01:27:56 2014`
	Example:
	`switch# install activate n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin Install operation 2 !!WARNING!! This patch will get activated only after a reload of the switch. at Sun Mar 9 00:42:12 2014`
		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.)
		Note
		Press **?** after a partial package name to display all possible matches available for activation. If there is only one match, press the **Tab** key to fill in the rest of the package name.

Step 4	Repeat Step 3 until all packages are activated.	Activates additional packages as required.

	Command or Action	Purpose
Step 5	(Optional) **show install active** **Example:** `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.

SUMMARY STEPS

1. **install commit** *filename*
2. (Optional) **show install committed**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**install commit** *filename* **Example:** `switch# install commit n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin`	Commits the current set of packages so that these packages are used if the device is restarted.
Step 2	(Optional) **show install committed** **Example:** `switch# show install committed`	Displays which packages are 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.

SUMMARY STEPS

1. **install deactivate** *filename*
2. (Optional) **show install inactive**
3. (Optional) **install commit**
4. (Optional) **install remove** {*filename* | **inactive**}

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**install deactivate** *filename* **Example:** `switch# install deactivate n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin`	Deactivates a package that was added to the device and turns off the package features for the line card. **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** **Example:** `switch# show install inactive`	Displays the inactive packages on the device.
Step 3	(Optional) **install commit** **Example:** `switch# install commit`	Commits the current set of packages so that these packages are used if the device is restarted. **Note** Packages can be removed only if the deactivation operation is committed.

Step 4	(Optional) **install remove** {*filename* | **inactive**}
	Example:
	`switch# install remove n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin Proceed with removing n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin? (y/n)? [n] y`
	Example:
	`switch# install remove inactive Proceed with removing? (y/n)? [n] y`
		Removes the inactive package.
		• Only inactive packages can be removed.
		• Packages can be removed only if they are deactivated from all line cards in the device.
		• The package deactivation must be committed.
		• To remove a specific inactive package from a storage device, use the **install remove** command with the *filename* argument.
		• To remove all inactive packages from all nodes in the system, use the **install remove** command with the **inactive** keyword.

Displaying Installation Log Information

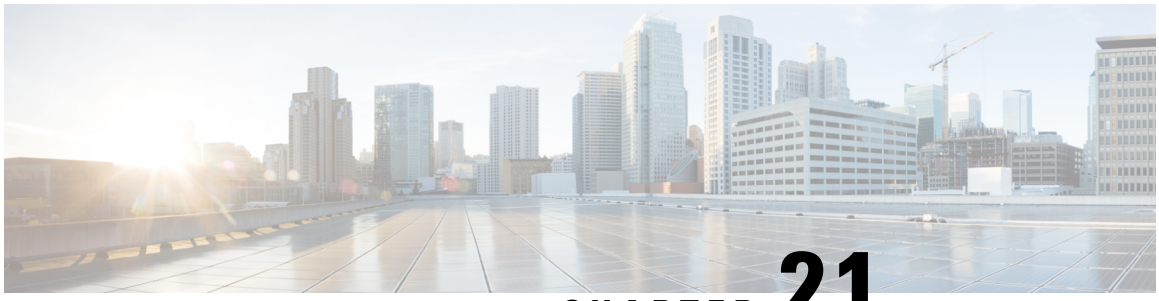
The installation log provides information on the history of the installation operations. Each time an installation operation is run, a number is assigned to that operation.

- 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
Thu Jan 9 01:26:09 2014
Install operation 1 by user 'admin' at Thu Jan 9 01:19:19 2018
Install add bootflash: n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin
Install operation 1 completed successfully at Thu Jan 9 01:19:24 2014
---------------------------------------
Install operation 2 by user 'admin' at Thu Jan 9 01:19:29 2018
Install activate n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin
Install operation 2 completed successfully at Thu Jan 9 01:19:45 2018
---------------------------------------
Install operation 3 by user 'admin' at Thu Jan 9 01:20:05 2018
Install commit n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin
Install operation 3 completed successfully at Thu Jan 9 01:20:08 2018
---------------------------------------
Install operation 4 by user 'admin' at Thu Jan 9 01:20:21 2018
Install deactivate n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin
Install operation 4 completed successfully at Thu Jan 9 01:20:36 2018
---------------------------------------
Install operation 5 by user 'admin' at Thu Jan 9 01:20:43 2018
Install commit n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin
Install operation 5 completed successfully at Thu Jan 9 01:20:46 2014
---------------------------------------
Install operation 6 by user 'admin' at Thu Jan 9 01:20:55 2018
Install remove n3500-uk9.6.0.2.U6.0.1.CSCab00001.bin
Install operation 6 completed successfully at Thu Jan 9 01:20:57 2018
```



CHAPTER 21

Configuration Replace and Commit-timeout

- Overview, on page 247
- Guidelines and Limitations, on page 256
- Perform Configuration Replace, on page 260

Overview

Configuration Replace

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.

Commit-Timeout

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.

Operation Steps

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.

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.

Table 31: DIfference 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

- 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.

Recommended Workflow



Note
- This workflow needs to be the same in the candidate config.
- Default configuration in the candidate config is not supported.

1. Generate a configuration file by first applying the configurations on a Cisco Nexus Series device and then use the **show running-configuration** output as the configuration file. Use this file to make configuration modifications as required. Then use this generated or updated configuration file to perform configuration replace.
2. View and verify the patch file by executing the **configure replace** <*file*> **show-patch** command. This is an optional step.
3. Run the configuration replace file either using or skipping the **commit-timeout** <*time*> feature. Based on your requirements, you can perform one of the following steps:
   - Run **configure replace** <*file*> **verbose** to see the commands that get executed with configuration replace on the console.
   - Run the **configure replace** [**bootflash**/**scp**/**sftp**] <*user-configuration-file*> **verbose commit-timeout** <*time*> commands to configure the commit time.
4. Run the **configure replace commit** command to stop the commit timer. This step is necessary if you have run the configuration replace operation with the commit-timeout feature.
5. Configuration replace performs a precheck that includes the semantic validation of the configuration. The configuration replace operation fails if there is an error. Use the **show config-replace log verify** command to see the details of the failed configurations. After applying the patch file, configuration replace triggers the verification process. The configuration replace compares the running-configuration with the user configuration file during the verification process. If there is a mismatch, it restores the device to the original configuration. Use the **show config-replace log verify** command to see the mismatched configurations.
6. You can perform the following configuration replace operations in Cisco NX-OS Release 9.3(1):
   - Configuration replace without the semantic validation and without best-effort mode.
   - Configuration replace without the semantic validation and with best-effort mode.

- Configuration replace with the semantic validation and without best-effort mode.
- Configuration replace with the semantic validation and with best-effort mode.

Verification

To check and verify configuration replace and its status, use the commands that are outlined in the table:

Table 32: Verifying Configuration Replace

Command	Purpose
configure replace [**bootflash**/**scp**/**sftp**]<*user-configuration-file*] **show-patch**	Displays the difference between the running-configurations and user-provided configurations.
show config-replace log exec	Displays a log of all the configurations executed and those that failed. In case of an error, it displays an error message against that configuration.
show config-replace log verify	Displays the configurations that failed, along with an error message. It does not display configurations that were successful.
show config-replace status	Displays the status of the configuration replace operations, including in-progress, successful, and failure. If you have configured the commit-timeout feature, the commit and timer status and the commit timeout time remaining is also displayed.

Configuration Examples

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)#
```

Configure Replace might fail when the manually created configuration is used instead of the configuration generated from the switch. The reason for possible failures is the potential difference in the default configuration that isn't shown in the show running configuration. Refer to the following examples:

If the power redundant command is the default command, it doesn't get displayed in the default configuration. But it's displayed when you use the **show run all** command. See the following example:

```
switch# show run all

!Command: show running-config all
!Running configuration last done at: Tue Nov 12 11:07:44 2019
!Time: Tue Nov 12 11:16:09 2019

version 9.3(1) Bios:version 05.39
power redundancy-mode ps-redundant
no hardware module boot-order reverse
no license grace-period
<snip>
hostname n9k13
```

The power redundant command isn't shown in the show running configuration command out. See the following example:

```
!Command: show running-config
!Running configuration last done at: Tue Nov 12 11:07:44 2019
!Time: Tue Nov 12 11:17:24 2019

version 9.3(1) Bios:version 05.39
hostname n9k13
```

When the **power redundancy-mode ps-redundant** command is added in the user configuration for the configure replace; then the verification/commit might fail. See the following example:

```
switch# show file bootflash:test

!Command: show running-config
!Running configuration last done at: Tue Nov 12 10:56:49 2019
!Time: Tue Nov 12 11:04:57 2019

version 9.3(1) Bios:version 05.39
power redundancy-mode ps-redundant
hostname n9k13
```

The **power redundancy-mode ps-redundant** command will not be shown in the show running after configure replace; therefore it will be considered as “missing” and the CR will fail. An example is given below.

```
switch# config replace bootflash:test verify-and-commit

Version match between user file and running configuration.
Pre-check for User config PASSED
ADVISORY: Config Replace operation started...
Modifying running configuration from another VSH terminal in parallel
is not recommended, as this may lead to Config Replace failure.

Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Collecting Running-Config
.Generating Rollback Patch

Validating Patch
```

```
Patch validation completed successful
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Rollback Patch
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Patch for verification
Verification failed, Rolling back to previous configuration
Collecting Running-Config
Cleaning up switch-profile buffer
Generating Rollback patch for switch profile
Executing Rollback patch for switch profiles. WARNING - This will change the
configuration of switch profiles and will also affect any peers if configured
Collecting Running-Config
Generating Rollback Patch
Rollback Patch is Empty
Rolling back to previous configuration is successful

Configure replace failed. Use 'show config-replace log verify' or 'show config-replace
log exec' to see reasons for failure

n9k13# show config-replace log verify
Operation : Config-replace to user config
Checkpoint file name : .replace_tmp_31849
Scheme : tmp
Cfg-replace done By : agargula
Cfg-replace mode : atomic
Verbose : disabled
Start Time : Tue, 11:20:59 12 Nov 2019
Start Time UTC : Tue, 10:20:59 12 Nov 2019
------------------------------------------
End Time : Tue, 11:21:28 12 Nov 2019
End Time UTC : Tue, 10:21:28 12 Nov 2019
Status : Failed

Verification patch contains the following commands:
-------------------------------------------------
!!
Configuration To Be Added Missing in Running-config
=================================================
!
power redundancy-mode ps-redundant

Undo Log
--------------------------------------------------------------------------------
End Time : Tue, 11:21:32 12 Nov 2019
End Time UTC : Tue, 10:21:32 12 Nov 2019
Status : Success
n9k13#
```

In the above example, CR will consider the default commands that are missing and will therefore fail.

Guidelines and Limitations

General Guidelines

- 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.
- 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.
- For a successful configuration replace operation, sequence number must be present for all ACE entries in ACL in the target configuration file.
- 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.
- If the line card to which the configuration replace feature is applied is offline, the configuration replace operation fails.
- An ITD service must be shut down (**shutdown**) prior to making ITD changes with the configuration replace feature.
- Using the **configure replace** command from maintenance mode asks for a user-confirmation with the following warning:

```
Warning: System is in maintenance mode. Please ensure user config won't inadvertently
revert back config in maintenance mode profile.
            Do you wish to proceed anyway? (y/n)  [n]
```

- Sequence number is mandatory for CLI **ip community-list** and **ip as-path access-list** commands. Without a sequence number, the configuration replace operation fails.
- If your configurations demand reloading the Cisco NX-OS device in order to apply the configuration, then you must reload these configurations after the configuration replace operation.

Configuration Replace for PBR Commands

The content of this section is applicable from Cisco NX-OS Release 10.4(3)F.

None of the PBR commands can coexist under the same parent route-map. If the mutually exclusive PBR commands are given under the same route-map in the candidate config, the config-replace patch is generated only for the last command variant under the route-map and is applied after CR operation.

The following table depicts a few use cases.

Use Case	Candidate Config	Converted Candidate Config
Use Case 1: Multiple command variants - Only the last command variant is retained The candidate config is automatically converted as shown in the third column before the CR patch is generated.	route-map rmap1 permit 10 set ip next-hop 1.1.1.1 2.2.2.2 set ipv6 next-hop 3::3 set ip next-hop verify-availability 4.4.4.4 set ip next-hop verify-availability 5.5.5.5 set ip vrf green next-hop 6.6.6.6 set ip vrf blue next-hop 7.7.7.7 8.8.8.8	route-map rmap1 permit 10 set ip vrf green next-hop 6.6.6.6 set ip vrf blue next-hop 7.7.7.7 8.8.8.8
Use Case 2: Commands comprising track IDs - Only the last command variant with same next-hop and different track ID is retained For the verify-availability commands, track ID cannot be modified for the same next-hop. The candidate config is automatically converted as shown in the third column before the CR patch is generated.	route-map test permit 10 set ip next-hop verify-availability 1.1.1.1 track 1 set ip next-hop verify-availability 2.2.2.2 track 20 set ip next-hop verify-availability 2.2.2.2 track 30 set ip next-hop verify-availability 2.2.2.2 track 40 set ip next-hop verify-availability 3.3.3.3 track 3	route-map test permit 10 set ip next-hop verify-availability 1.1.1.1 track 1 set ip next-hop verify-availability 2.2.2.2 track 40 set ip next-hop verify-availability 3.3.3.3 track 3

Commands with Mandatory Sequence Number

Ensure that you provide the sequence number mandatorily for the following commands in the candidate-config file. Without a sequence number the configuration replace operation fails:

- **ip prefix-list list-name seq** *seq* { **deny** | **permit** } *prefix*
- **ipv6 prefix-list list-name seq** *seq* { **deny** | **permit** } *prefix*
- **mac-list list-name seq** *seq* { **deny** | **permit** } *prefix*
- **ip community-list** { **standard** | **expanded** } *list-name* seq *seq* { **deny** | **permit** } *expression*
- **ip extcommunity-list** { **standard** | **expanded** } *list-name* seq *seq* { **deny** | **permit** } *expression*
- **ip large-community-list** { **standard** | **expanded** } *list-name* seq *seq* { **deny** | **permit** } *expression*
- **ip-as-path access-list** *list-name* seq *seq* { **deny** | **permit** } *expression*

Not Supported Features

1. 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.
2. The configuration replace operation is not supported if you attempt to replace a multichassis EtherChannel trunk (MCT) configuration with a virtual peer-link configuration. This operation is not allowed because

the physical MCT uses the CFS distribution over Ethernet mode and the virtual peer-link use the CFS distribution over IP mode.

3. Entering maintenance mode from the user configuration is not supported.
4. The semantic validation is not supported in 4-Gig memory platforms.
5. The configuration replace feature does not support features that require a reload. One such feature is: system vlan reserve.
6. The configuration replace feature does not work if the FEX line card is offline.
7. The configuration replace feature does not support autoconfigurations.

Supported Features

1. The configuration replace feature is supported on Cisco Nexus 3000 Series and Cisco Nexus 9000 Series switches.
2. Using the **configure replace** command from maintenance mode with a *<non-interactive>* option is supported. It takes the *yes* user-confirmation by default and proceeds.
3. The configuration replace feature is supported only for the configure terminal mode commands. The configure profile, configure jobs, and any other modes are not supported.
4. Beginning with Cisco NX-OS Release 9.3(4), the configuration replace feature is supported for breakout interface configurations.

Recommended Guidelines

1. 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.
2. 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.

Configuration Guidelines

1. 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.
2. In the verify-only mode, the TCAM-dependent configuration may not throw an error and gets succeeded. However, it may fail during actual CR operation. To avoid this, it is recommended to apply TCAM carving configuration and reload before performing CR.
3. Replay of configuration generated from one device and applied on another device goes through and CLI CR is successful but SNMPv3 operations fail with 'Authentication failure' message in the new device.
4. 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.

5. 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.
6. 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.
7. 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.
8. Clear text passwords are allowed in the case of configuration replace candidate-config file as the user database gets synced between SNMP and AAA (Security).

Release Specific Guidelines

- Beginning with Cisco NX-OS Release 9.3(5) the configuration replace feature is supported on port profiles. The configure jobs mode is supported. Configuration files with scheduler job commands can be used for configuration replace. Configuration replace (CR) for FEX interface configurations is supported. Provisioning of FEX is not supported through CR. Once provisioned, configurations on the FEX interfaces can modified through CR.



Note This guideline does not apply to Cisco Nexus 3000-series platform switches, on which FEX is not supported.

- Beginning with Cisco NX-OS Release 9.3(6), the **boot nxos** *image* configuration can be excluded in the **show running-config** , **show startup-config** , **copy running-config** *filename* , and **copy startup-config** *filename* commands by configuring **service exclude-bootconfig** .
- Beginning with NX-OS Release 10.3(1)F, the configuration replace feature does not support feature app-hosting.
- Beginning from Cisco NX-OS Release 10.4(2)F, the configuration replace feature is supported for LDAP on Cisco NX-OS devices.
- Beginning from Cisco NX-OS Release 10.4(2)F, for non-case sensitive commands, if there is a letter case distinction between the commands in running config and candidate-config files, then the output of **config replace show-patch** displays both the commands due to the difference in letter case.
- Beginning from Cisco NX-OS Release 10.4(3)F, you can also use polymorphic commands in candidate configuration to perform configuration replace.
- Beginning with Cisco NX-OS Release 10.5(1)F, the following cannot be part of the same CR candidate file:
  - **no hardware access-list update atomic**
  - ACL configuration that exceeds existing running-configuration atomic TCAM configuration limits
- Beginning with Cisco NX-OS Release 10.5(1)F, sequence number is mandatory for vlan access-map command. Without a sequence number the configuration replace operation fails.

Perform Configuration Replace

Replace the running configuration on the switch with a candidate or user-provided configuration file using the **configure replace** command.

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

Step 1 Perform configuration replace using the command **configure replace** { < *uri_local* > | < *uri_remote* > } [**verbose** | **show-patch**]

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 Display the differences between the running-configuration and the user-provided configuration using the command **configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **show-patch**

Note
- This command does not encrypt plain text password.
- This command can still show a patch even after configuration replace is successful for CLI snmp-server traps commands .

Step 3 Replace the configuration on the switch with the new user configuration that is provided by the user using the command **configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **verbose**

Configuration replace is always atomic.

Step 4 Replace the configuration on the switch with the new user configuration and enables the configuration replace with semantic validation using the command **configure replace** <*user-configuration-file*> [**best-effort**]

The best-effort option enables the configuration replace to execute the full patch despite any error in the commands and also make sure that the previous configuration is not rolled back.

Beginning with Cisco NX-OS Release 10.5(1)F, configuration replace feature supports batch ACL configurations on Cisco Nexus 9300-FX2/FX3/GX Series switches. If the **best effort** mode is enabled, any failure within the batched configuration will result in skipping the entire set of configurations in that particular batch.

Step 5 Replace the configuration on the switch with the new user configuration and enables the configuration replace with semantic validation using the command **configure replace** <*user-configuration-file*> [**verify-and-commit**]

The verify-and-commit option is used for enabling the semantic validation. Patch will be executed only if semantic validation of the full patch gets passed.

You can use the best-effort option or the verify-and-commit option or both the options at the same time.

Step 6 Show only the patch and does Semantic validation on the patch, and display the results using the command **configure replace** <*user-configuration-file*> [**verify-only**]

The patch does not get applied to the system.

Step 7 Configure the commit time in seconds using the command **configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **verbose commit-timeout** <*time*>

This is an optional step. The timer starts after the configuration replace operation is successfully completed.

Step 8 Stop the commit timer and continues the configuration replace configuration using the command **configure replace** [**commit**] and use the command **configure replace** [**bootflash** / **scp** / **sftp**] <*user-configuration-file*> *non-interactive* for maintainence mode.

This is an optional step.

Note
This step is applicable only if you have configured the commit-timeout feature.

Note
To rollback to the previous configuration, you must wait for the expiry of the commit timer. Once the timer expires, the switch is automatically rolled back to the previous configuration.

There is no user prompt in maintenance mode. The **yes** user-confirmation is taken by default, and rollback proceeds. You can use the non-interactive option only in the maintenance mode.

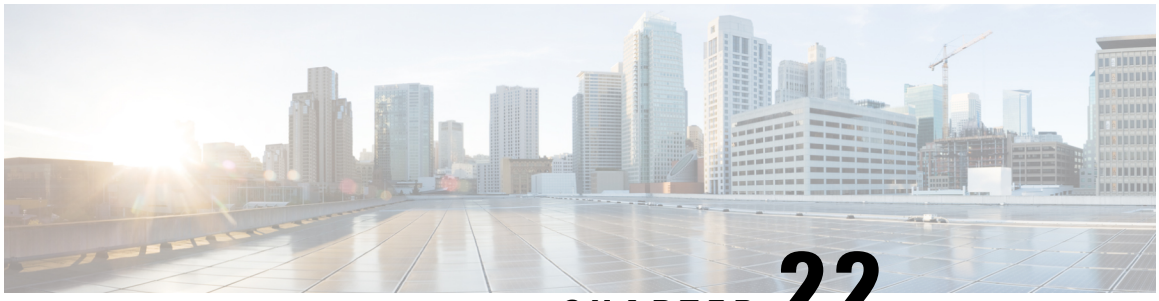
Step 9 (Optional) Reload the switch to the previously saved startup configuration if configure replace operation fails and the restore to the previous running configuration also fails using the command **configure replace** <*user-configuration-file*> [**on-failure reload** [**save-startup-config**]]

This is an optional step.

This option has no impact if:

- Configure replace is successful
- Configure replace fails, but restore to the previous running configuration is successful

Note
[**save-startup-config**] – This option saves the running config to startup before starting the configure replace operations.



CHAPTER 22

Configuring Rollback

This chapter contains the following sections:

- Information About Rollbacks, on page 263
- Guidelines and Limitations for Rollbacks, on page 263
- Creating a Checkpoint, on page 264
- Implementing a Rollback, on page 265
- Verifying the Rollback Configuration, on page 266

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.
- 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.

SUMMARY STEPS

1. switch# **checkpoint** { [*cp-name*] [**description** *descr*] |**file** *file-name*
2. (Optional) switch# **no checkpoint***cp-name*
3. (Optional) switch# **show checkpoint***cp-name*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **checkpoint** { [*cp-name*] [**description** *descr*] |**file** *file-name* **Example:** `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. The description can contain up to 80 alphanumeric characters, including spaces.
Step 2	(Optional) switch# **no checkpoint***cp-name* **Example:** `switch# no checkpoint stable`	You can use the **no** form of the **checkpoint** command to remove a checkpoint name. Use the **delete** command to remove a checkpoint file.

	Command or Action	Purpose
Step 3	(Optional) switch# **show checkpoint***cp-name* **Example:** [**all**] `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.

SUMMARY STEPS

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*}
2. **rollback running-config** {**checkpoint** *cp-name* | **file** *cp-file*} **atomic**

DETAILED STEPS

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*} **Example:** `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** **Example:** `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
```

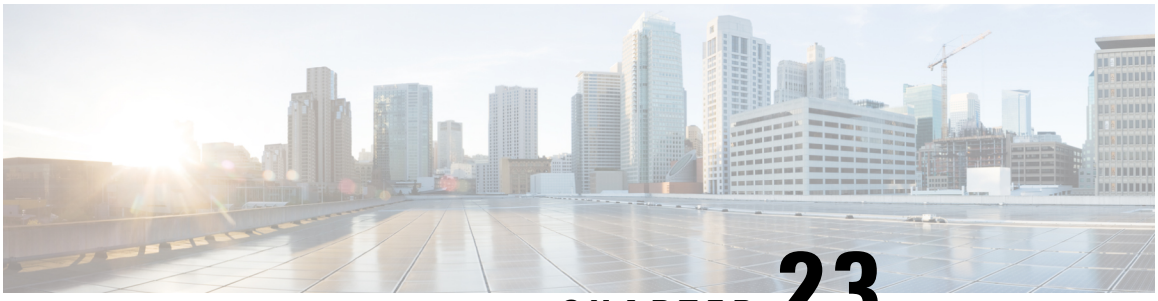
Verifying the Rollback Configuration

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 23

Candidate Configuration

- Overview, on page 267
- Guidelines and Limitations, on page 269
- Perform Integrity Check for Candidate Config, on page 274

Overview

Candidate configuration is a subset of the running-config which checks whether the Candidate config exists in the running-config without any additions or modifications or deletions.

To check the integrity of the candidate config, use the following commands:

- `show diff running-config`
- `show diff startup-config`

For more information on the CLIs, refer to Perform Integrity Check for Candidate Config, on page 274.

Examples

No Difference Between Running and Candidate Config

```
switch# show diff running-config bootflash:base_running.cfg
switch#
```

Difference Between Running and Candidate

```
switch# show diff running-config bootflash:modified-running.cfg unified
--- running-config
+++ User-config
@@ -32,11 +32,11 @@

interface Ethernet1/1
   mtu 9100
   link debounce time 0
   beacon
-  ip address 2.2.2.2/24
+  ip address 1.1.1.1/24
   no shutdown

interface Ethernet1/2

interface Ethernet1/3
switch#
```

Difference Between Running and Partial Candidate

```
switch# show file bootflash:intf_vlan.cfg
interface Vlan101
  no shutdown
  no ip redirects
  ip address 1.1.2.1/24 secondary
  ip address 1.1.1.1/24
switch#
switch# show diff running-config bootflash:intf_vlan.cfg partial unified
--- running-config
+++ User-config
@@ -3897,10 +3883,14 @@
   mtu 9100
   ip access-group IPV4_EDGE in
   ip address 2.2.2.12/26 tag 54321

 interface Vlan101
+ no shutdown
+ no ip redirects
+ ip address 1.1.2.1/24 secondary
+ ip address 1.1.1.1/24

 interface Vlan102
   description Vlan102
   no shutdown
   mtu 9100
switch#
```

Partial Configuration Diff Merged

```
switch# show file po.cfg
interface port-channel500
description po-123
switch#
switch# sh run int po500

!Command: show running-config interface port-channel500
!Running configuration last done at: Fri Sep 29 12:27:28 2023
!Time: Fri Sep 29 12:30:24 2023

version 10.4(2) Bios:version 07.69

interface port-channel500
  ip address 192.0.2.0/24
  ipv6 address 2001:DB8:0:ABCD::1/48

switch#

switch# show diff running-config po.cfg partial merged unified
--- running-config
+++ User-config
@@ -124,10 +110,11 @@
interface port-channel100
 interface port-channel500
   ip address 192.0.2.0/24
   ipv6 address 2001:DB8:0:ABCD::1/48
+ description po-123
 interface port-channel4096
 interface Ethernet1/1
switch#
```

Guidelines and Limitations

- The line numbers that are displayed in the generated running config do not match with the candidate config as they are internally generated one.
- If there is any difference between the configuration of running and candidate, then it is displayed inline as output.
- If the whole block of configuration in the candidate file is a new addition, it will be appended at the end of the generated running config.
- When the candidate config has an SNMP or an AAA user CLI with clear-text password, the SNMP user is seen as a diff even when the user is already configured.
- EIGRP address family IPv4 configs are recommended to configure under the EIGRP address family hierarchy and not under the router mode hierarchy in the candidate file, before running a partial diff.
- If the target/candidate file has a default command (for example, `- log-neighbor-warnings;`) configured directly under the **router eigrp** mode and not one of its submodes, that is, **address-family ipv4 unicast** or **address-family ipv6 unicast** , then partial-diff shows + displayed in the output of the default command (for example, `+ log-neighbor-warnings`) in the diff.
- For noncase sensitive commands, if there is a letter case distinction between the commands in the running config and candidate-config files, then the output of **partial diff** displays both the commands due to the difference in letter case.
- Cleartext passwords are allowed in case of partial diff candidate CONFIG_FILE as the user database gets synced between SNMP and AAA (Security).
- Configuration profile, maintenance profile (mmode) and scheduler mode configurations are not supported.

Integrity Check Guidelines

1. To perform an integrity check on a full running configuration file, the user must supply a running configuration generated by an NX-OS system and should not use the **partial** keyword. The **partial** keyword treats the input candidate as a subset of the running configuration.

Release Specific Guidelines

1. Beginning with Cisco NX-OS Release 10.2(3)F, Candidate config integrity check option is introduced on all Cisco Nexus switches.
2. Beginning from Cisco NX-OS Release 10.4(3)F, you can also use polymorphic commands in candidate configuration to perform partial diff.

Partial Diff of Default Commands for Multicast Components Guidelines

The content of this section is applicable from Cisco NX-OS Release 10.4(3)F.

If the default commands of multicast components are present in the candidate CONFIG_FILE, they are seen in show diff as follows:

Multicast Component	Default Commands in show diff
PIM	ip access-list copp-system-p-acl-pim 10 permit pim any 224.0.0.0/24 20 permit udp any any eq pim-auto-rp ip access-list copp-system-p-acl-pim-mdt-join ip access-list copp-system-p-acl-pim-reg 10 permit pim any any
PIM6	ipv6 access-list copp-system-p-acl-pim6 10 permit pim any ff02::d/128 20 permit udp any any eq pim-auto-rp ipv6 access-list copp-system-p-acl-pim6-reg 10 permit pim any any
IGMP	ip access-list copp-system-p-acl-igmp 10 permit igmp any 224.0.0.0/3 class-map copp-system-p-class-normal-igmp
MLD	ipv6 access-list copp-system-p-acl-mld 10 permit icmp any any mld-query 20 permit icmp any any mld-report 30 permit icmp any any mld-reduction 40 permit icmp any any mldv2

show diff running-config *file_url* [unified] [partial] [merged] Command Guidelines

- When using the **unified** , **partial** , and **merged** option to review the differences for the following PBR commands, the diff outputs are as mentioned below:
  - **set ip next-hop**
  - **set ip default next-hop**
  - **set ip default vrf next-hop**
  - **set ipv6 next-hop**
  - **set ipv6 default next-hop**
  - **set ipv6 default vrf next-hop**

1. If the candidate next-hops are a subset of running next-hops (in the same order and sequence), and candidate additive flags are a subset of running flags, then the diff output is empty as shown in the following table:

Candidate Config	Running Config	Partial Unified Merged Diff Output
route-map rmap1 permit 10 set ip next-hop 1.1.1.1 2.2.2.2 load-share	route-map rmap1 permit 10 set ip next-hop 1.1.1.1 2.2.2.2 3.3.3.3 load-share force-order	<no-diff>

2. If the candidate next-hops are a subset of running next-hops (in the same order and sequence), and the candidate has some extra additive flags which are not present in running config, then the diff output appends any additional flags present in the candidate config to the running config, similar to command line behavior as shown in the following table:

Candidate Config	Running Config	Partial Unified Merged Diff Output
route-map rmap1 permit 10 set ip next-hop 1.1.1.1 2.2.2.2 load-share force-order	route-map rmap1 permit 10 set ip next-hop 1.1.1.1 2.2.2.2 3.3.3.3 load-share drop-on-fail	route-map rmap1 permit 10 - set ip next-hop 1.1.1.1 2.2.2.2 3.3.3.3 load-share drop-on-fail + set ip next-hop 1.1.1.1 2.2.2.2 3.3.3.3 load-share force-order drop-on-fail

3. If candidate next-hops are not a subset of running next-hops (in the same order and sequence), and the candidate and running record can have any additive flag, then the diff output indicates this with a '–' for the running config record and a '+' for the candidate config record.

   This distinction is important, particularly when using with PBR commands, where the sequence of next-hops is critical. Even if the next-hops IP addresses are identical, their order affects functionality.

   For example, '1.1.1.1 2.2.2.2' is different from '2.2.2.2 1.1.1.1'.



Important If there is an additive flag in the running config that you wish to retain after merging with the candidate config, you must explicitly include that flag in the candidate config. This ensures that the needed flags are preserved in the final, merged configuration.

Candidate Config	Running Config	Partial Unified Merged Diff Output
route-map rmap1 permit 10 set ip next-hop 1.1.1.1 2.2.2.2 load-share drop-on-fail	route-map rmap1 permit 10 set ip next-hop 2.2.2.2 1.1.1.1 load-share force-order	route-map rmap1 permit 10 - set ip next-hop 2.2.2.2 1.1.1.1 load-share force-order + set ip next-hop 1.1.1.1 2.2.2.2 load-share drop-on-fail

- When **Partial Unified** or **Partial Unified Merged** option is used, all the PBR commands are mutually exclusive and cannot coexist within the same parent route-map. Therefore, if a candidate configuration specifies multiple mutually exclusive PBR commands under a single route-map, only the last command variant will be shown in the partial diff output.

  Example-1: In this example, the candidate configuration includes multiple PBR commands under a single route-map **rmap1** :

```
route-map rmap1 permit 10
  set ip next-hop 1.1.1.1 2.2.2.2
  set ipv6 next-hop 3::3
  set ip next-hop verify-availability 4.4.4.4
  set ip next-hop verify-availability 5.5.5.5
  set ip vrf green next-hop 6.6.6.6
  set ip vrf blue next-hop 7.7.7.7 8.8.8.8
```

  Before the generation of the partial-diff output, the above candidate configuration is automatically converted to the following:

```
route-map rmap1 permit 10
  set ip vrf green next-hop 6.6.6.6
  set ip vrf blue next-hop 7.7.7.7 8.8.8.8
```

Example-2: In this example, the candidate configuration includes multiple 'set ip next-hop verify-availability' commands with different track IDs specified for the route-map **rmap2** . Since track IDs cannot be modified for the same next-hop, these commands are mutually exclusive:

```
route-map rmap2 permit 10
  set ip next-hop verify-availability 1.1.1.1 track 1
  set ip next-hop verify-availability 2.2.2.2 track 20
  set ip next-hop verify-availability 2.2.2.2 track 30
  set ip next-hop verify-availability 2.2.2.2 track 40
  set ip next-hop verify-availability 3.3.3.3 track 3
```

Before generating the partial-diff output, the system will automatically consolidate these commands by retaining only the last **set ip next-hop verify-availability** command for each next-hop IP address as shown below:

```
                                route-map rmap2 permit 10
  set ip next-hop verify-availability 1.1.1.1 track 1
                                set ip next-hop verify-availability 2.2.2.2 track 40
  set ip next-hop verify-availability 3.3.3.3 track 3
```

- When the **Partial Unified Merged** option is used, to review the differences for the **verify-availability** command variants, the track ID for a given next-hop is not modifiable.

  Therefore, if the candidate and running configurations contain the same next-hop but have different track IDs under the same parent route-map, the candidate record cannot simply be merged with the running record, as per command line behavior. Therefore, to apply the candidate record with different track ID for the same next-hop, the corresponding running config record must be removed first ('–' for the running configuration record in the diff) and then when the candidate record is merged, it will be appended at the end of the last record under the same parent route-map ('+' for the candidate config record).

  The following table shows the sample candidate and running configuration with the 'Partial Unified Merged' output for different use cases as mentioned below:

  1. If the track ID is different for the same next-hop under candidate and running config, then the diff output is as shown in the following table:

Candidate Config	Running Config	Partial Unified Merged Diff Output

route-map rmap1 permit 10
  set ip next-hop
verify-availability 1.1.1.1
 track 1
  set ip next-hop
verify-availability 2.2.2.2
 track 20
  set ip next-hop
verify-availability 3.3.3.3
 track 3 load-share
	route-map rmap1 permit 10
	  set ip next-hop
	verify-availability 1.1.1.1
	 track 1
	  set ip next-hop
	verify-availability 2.2.2.2
	 track 2
	  set ip next-hop
	verify-availability 3.3.3.3
	 track 3 load-share
		route-map test permit 10
		  set ip next-hop
		verify-availability 1.1.1.1
		track 1
		 - set ip next-hop
		verify-availability 2.2.2.2
		track 2
		  set ip next-hop
		verify-availability 3.3.3.3
		track 3
		 + set ip next-hop
		verify-availability 2.2.2.2
		track 20 load-share

  2. If track ID is not present under candidate config but present in running config for the same next-hop, then the diff output is empty as shown in the following table:

Candidate Config	Running Config	Partial Unified Merged Diff Output

route-map rmap1 permit 10
  set ip next-hop
verify-availability 1.1.1.1
 track 1
  set ip next-hop
verify-availability 2.2.2.2
 	
  set ip next-hop
verify-availability 3.3.3.3
 track 3
	route-map rmap1 permit 10
	  set ip next-hop
	verify-availability 1.1.1.1
	 track 1
	  set ip next-hop
	verify-availability 2.2.2.2
	 track 2
	  set ip next-hop
	verify-availability 3.3.3.3
	 track 3
		no-diff

3. If track ID is not present under running config but present in candidate config for the same next-hop, then the diff output is as shown in the following table:

Candidate Config	Running Config	Partial Unified Merged Diff Output

route-map rmap1 permit 10
  set ip next-hop
verify-availability 1.1.1.1
 track 1
  set ip next-hop
verify-availability 2.2.2.2
 track 20
  set ip next-hop
verify-availability 3.3.3.3
 track 3
	route-map rmap1 permit 10
	  set ip next-hop
	verify-availability 1.1.1.1
	 track 1
	  set ip next-hop
	verify-availability 2.2.2.2
	
	  set ip next-hop
	verify-availability 3.3.3.3
	 track 3
		route-map rmap1 permit 10
		  set ip next-hop
		verify-availability 1.1.1.1
		track 1
		 - set ip next-hop
		verify-availability 2.2.2.2
		  set ip next-hop
		verify-availability 3.3.3.3
		track 3
		 + set ip next-hop
		verify-availability 2.2.2.2
		track 20

Partial Diff of RPM Commands Guidelines

The content of this section is applicable from Cisco NX-OS Release 10.4(3)F.

When using the unified, partial, and merged option to review the differences for the following RPM commands, the diff outputs are as follows:

- In the candidate configuration, the RPM commands will undergo syntactic validation as reflected in the diff output. However, semantic validation will not be performed in the diff output. It is the user's responsibility to ensure that the commands in the candidate configuration are semantically accurate.

  If the command in the Candidate-config is semantically incorrect, the diff may incorrectly indicate that the command is executable, but in actual it may not.

- Ensure that you provide the sequence number mandatorily for the following commands in the Candidate-config file:
  - **ip prefix-list list-name seq** *seq* { **deny** | **permit** } *prefix*
  - **ipv6 prefix-list list-name seq** *seq* { **deny** | **permit** } *prefix*
  - **mac-list list-name seq** *seq* { **deny** | **permit** } *prefix*
  - **ip community-list** { **standard** | **expanded** } *list-name* seq *seq* { **deny** | **permit** } *expression*
  - **ip extcommunity-list** { **standard** | **expanded** } *list-name* seq *seq* { **deny** | **permit** } *expression*
  - **ip large-community-list** { **standard** | **expanded** } *list-name* seq *seq* { **deny** | **permit** } *expression*

- **ip-as-path access-list** *list-name* seq *seq* { **deny** | **permit** } *expression*

- When the following commands include an expression string that has spaces enclosed in quotes within the Candidate-config, there will be no differences displayed in the diff output:
  - **ip community-list expanded** *list-name* seq *seq* { **deny** | **permit** } *expression*
  - **ip extcommunity-list expanded** *list-name* seq *seq* { **deny** | **permit** } *expression*
  - **ip large-community-list expanded** *list-name* seq *seq* { **deny** | **permit** } *expression*
  - **ip-as-path access-list** *list-name* seq *seq* { **deny** | **permit** } *expression*

Candidate Config	Running Config	Partial Unified [Merged] Diff Output
`ip community-list expanded list_abc seq 10 permit "1:1 "`	`ip community-list expanded list_abc seq 10 permit "1:1"`	`no-diff`
`ip extcommunity-list expanded list_abc seq 10 permit "1:1  "`	`ip extcommunity-list expanded list_abc seq 10 permit "1:1"`	`no-diff`
`ip large-community-list expanded list_abc seq 10 permit "1:1:1   "`	`ip large-community-list expanded list_abc seq 10 permit "1:1:1"`	`no-diff`
`ip as-path access-list list_abc seq 10 permit "1 "`	`ip as-path access-list list_abc seq 10 permit "1"`	`no-diff`

Perform Integrity Check for Candidate Config

To perform the integrity check, use the following commands:

Before you begin

Note Before performing the integrity check, ensure that the running config and the candidate config belong to the same image version.

Procedure

Step 1 Display the differences between the running and user given candidate config using the command **show diff running-config** *file_url* [**unified**] [**merged**]

Example:

```
switch# show diff running-config bootflash:candidate.cfg partial unified
```

- *file_url* : File path to compare with.

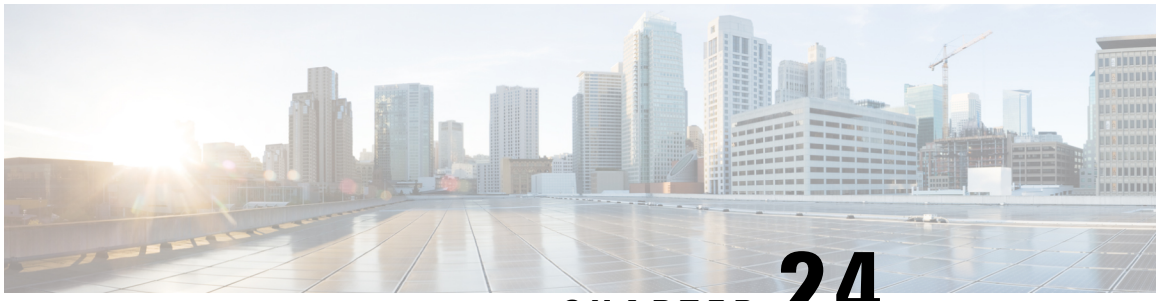
- **unified** : Displays the difference between running and user configuration in unified format.
- **merged** : Enter **merged** only if sub-commands need to be merged instead of replace.

Step 2 Display the differences between the startup and user given candidate config using the command **show diff startup-config** *file_url* [**unified**]

Example:

```
switch# show diff startup-config bootflash:candidate.cfg unified
```

- *file_url* : File path to compare with.
- **unified** : Displays the difference between startup and user configuration in unified format.



CHAPTER 24

Configuring User Accounts and RBAC

This chapter contains the following sections:

- Information About User Accounts and RBAC, on page 277
- Guidelines and Limitations for User Accounts, on page 280
- Configuring User Accounts, on page 281
- Configuring RBAC, on page 282
- Verifying the User Accounts and RBAC Configuration, on page 286
- Configuring User Accounts Default Settings for the User Accounts and RBAC, on page 287

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. However, the network-operator role cannot run the **show running-config** and **show startup-config** commands.

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.

Note Only network-admin user can perform a Checkpoint or Rollback in the RBAC roles. Though other users have these commands as a permit rule in their role, the user access is denied when you try to execute these 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, VLANs, and VSANs.

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

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.

Note 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.
- 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.

SUMMARY STEPS

1. switch# **configure terminal**
2. (Optional) switch(config)# **show role**
3. switch(config) # **username** *user-id* [**password** *password*] [**expire** *date*] [**role** *role-name*]
4. switch(config) # **exit**
5. (Optional) switch# **show user-account**
6. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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. The *user-id* is a case-sensitive, alphanumeric character string with a maximum of 28 characters. The default *password* is undefined. **Note** If you do not specify a password, the user might not be able to log into the switch. The **expire** *date* option format is YYYY-MM-DD. The default is no expiry date.
Step 4	switch(config) # **exit**	Exists global configuration mode.

	Command or Action	Purpose
Step 5	(Optional) switch# **show user-account**	Displays the role configuration.
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
```

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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **role name** *role-name*
3. switch(config-role) # **rule** *number* {**deny** | **permit**} **command** *command-string*
4. switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**}
5. switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**} **feature** *feature-name*
6. switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**} **feature-group** *group-name*
7. (Optional) switch(config-role)# **description** *text*
8. switch(config-role)# **end**
9. (Optional) switch# **show role**
10. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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.

	Command or Action	Purpose
		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. The *command-string* can contain spaces and regular expressions. For example, interface ethernet * includes all Ethernet interfaces. Repeat this command for as many rules as needed.
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. Use the **show role feature** command to display a list of features. 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. Use the **show role feature-group** command to display a list of feature groups. 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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **role feature-group** *group-name*
3. switch(config) # **exit**
4. (Optional) switch# **show role feature-group**
5. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **role feature-group** *group-name*	Specifies a user role feature group and enters role feature group configuration mode. 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.

SUMMARY STEPS

1. switch# **configure terminal**

2. switch(config) # **role name** *role-name*
3. switch(config-role) # **interface policy deny**
4. switch(config-role-interface) # **permit interface** *interface-list*
5. switch(config-role-interface) # **exit**
6. (Optional) switch(config-role) # **show role**
7. (Optional) switch(config-role) # **copy running-config startup-config**

DETAILED STEPS

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. Repeat this command for as many interfaces as needed. For this command, you can specify Ethernet interfaces.
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.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **role name** *role-name*

3. switch(config-role)# **vlan policy deny**
4. switch(config-role-vlan # **permit vlan** *vlan-list*
5. switch(config-role-vlan) # **exit**
6. (Optional) switch# **show role**
7. (Optional) switch# **copy running-config startup-config**

DETAILED STEPS

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. 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.

Verifying the User Accounts and RBAC Configuration

Use one of the following commands to verify the configuration:

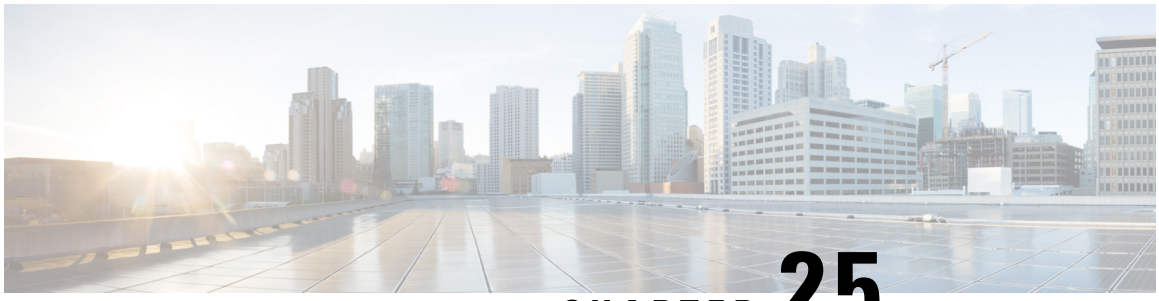
Command	Purpose
show role [*role-name*]	Displays the user role configuration
show role feature	Displays the feature list.
show role feature-group	Displays the feature group configuration.
show startup-config security	Displays the user account configuration in the startup configuration.
show running-config 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 33: 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.



CHAPTER 25

Configuring Secure Erase

- Information about Secure Erase, on page 289
- Prerequisites for Performing Secure Erase, on page 289
- Guidelines and Limitations for Secure Erase, on page 290
- Configuring Secure Erase, on page 290

Information about Secure Erase

Beginning with Cisco NX-OS Release 10.2(2)F, the Secure Erase feature is introduced to erase all customer information for Nexus 3548 switches. Secure Erase is an operation to remove all the identifiable customer information on Cisco NX-OS devices in conditions of product removal due to Return Merchandise Authorization (RMA), or upgrade or replacement, or system end-of-life.

Cisco Nexus 3548 switches consume storage to conserve system software images, switch configuration, software logs, and operational history. These areas can have customer-specific information such as details regarding network architecture, and design as well as a potential target for data thefts.

The Secure Erase process is used in the following two scenarios:

- Return Material Authorization (RMA) for a device - If you must return a device to Cisco for RMA, remove all the customer-specific data before obtaining an RMA certificate for the device.
- Recovering the compromised device - If the key material or credentials that are stored on a device is compromised, reset the device to factory configuration, and then reconfigure the device.

Note Secure Erase feature will not erase content in External storage.

The device reloads to perform a factory reset which results in the switch entering the power down mode. After a factory reset, the device clears all its environment variables including the **MAC_ADDRESS** and the **SERIAL_NUMBER** which are required to locate and load the software.

Prerequisites for Performing Secure Erase

- Ensure that all the software images, configurations, and personal data are backed up before performing the secure erase operation.

- Ensure that there is an uninterrupted power supply when the process is in progress.
- Ensure that neither In-Service Software Upgrade (ISSU) nor In-Service Software Downgrade (ISSD) is in progress before starting the secure erase process.

Guidelines and Limitations for Secure Erase

- FX3 or FX3S or FX3P switches are supported in TOR and FEX mode. If secure erase is done in FEX mode, a switch will boot in TOR mode after the secure erase operation.
- Software patches, if installed on the device, will not be restored after the Secure Erase operation.
- If the **factory-reset** command is issued through a session, the session is not restored after the completion of the factory reset process.

  The top of rack switches and supervisor modules returns to the loader prompt.

  End of row switch modules will be in a powered down state.

  If you configure secure erase of fex, the factory reset is initiated and fex configuration will be removed.

  Fex secure erase to be monitored using fex console. In case of failure, reboot and bring up fex and initiate secure erase again.

Configuring Secure Erase

To delete all necessary data before shipping to RMA, configure secure erase using the below command:

Command	Purpose
factory-resetfex module*mod* Example: `switch(config)# factory-reset [module <3>]`	Use the command with **all** options enabled. No system configuration is required to use the **factory reset** command. To secure erase for fex, use **factory-resetfex** [*allfex_no*] • To secure erase all fex at once, use option **all**. **Note** Ensure that the fex is not in Active-Active scenario, before initiating secure erase operation. Use the option **mod** to reset the start-up configurations: • For top of rack switches, the command is **factory-reset** or **factory-reset module 1**. • In LXC mode for top of rack switches, the command is **factory-reset module 1** or **27** • For end of row module switches, the command is **factory-reset module #module_number** After the factory reset process is successfully completed, the switch reboots and is powered down.

Note Parallel secure erase operations are not supported. To erase more than one module in single EoR chassis, the recommended order is line card, fabric, standby supervisor, system controller, and then active supervisor.

You can boot that secure erase image to trigger the data wipe.

The following is an example output for configuring secure erase factory reset command:

```
FX2-2-switch# factory-reset fex all
!!!! WARNING:
This command will perform factory-reset of all FEX modules !!!!
The factory reset operation will erase ALL persistent storage on the specified FEX module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable. Please, proceed with
caution and understanding that this operation cannot be undone and will leave the system
in a fresh-from-factory state.
!!!! WARNING !!!!

Do you want to continue? (y/n) [n] y
Initiating factory-reset for the FEX: 109 --- SUCCESS!!
FEX: 109 is reloading for the reset operation to proceed.
Factory reset may take time...
Please, wait and do not power off the FEX...
Trying to remove the FEX:109 config !!!
Initiating factory-reset for the FEX: 110 --- SUCCESS!!
FEX: 110 is reloading for the reset operation to proceed.
Factory reset may take time...
```

```
Please, wait and do not power off the FEX...
Trying to remove the FEX:110 config !!!
Successfully removed FEX:110 config. !!!
```

The following shows the example of fex logs:

```
FX2-2-switch# 2021
FEX console logs:
================
bgl-ads-4157:138> telnet 10.127.118.15 2007
Trying 10.127.118.15...
Connected to 10.127.118.15.
Escape character is '^]'.

fex-109#
fex-109# [129266.313614] writing reset reason 9, Factory-reset requested by abc
[129266.391801] Restarting system - Factory-reset requested by abc [9]
U-Boot 2011.12 (Jun 25 2014 - 16:28:41) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Golden image
U-boot retry count 0
Jump to upgradeable image at 0xefd20040
U-Boot 2011.12 (Jun 25 2014 - 16:19:54) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Upgradeable image
DRAM: Configuring DDR for 666.667 MT/s data rate
Time-out count = 480
DDR configuration get done
1 GiB (DDR3, 32-bit, CL=6, ECC on)
Memory test from 0x40000 to 0x1fdfffff
Data line test..... OK
Address line test..... OK
OK
Flash: 288 MiB
L2: 256 KB enabled
Set dbglevel to its default value (0x1)
PCIe1: Root Complex of mini PCIe SLOT, x1, regs @ 0xffe0a000
PCIe1: Bus 00 - 01
PCIe2: Root Complex of PCIe SLOT, no link, regs @ 0xffe09000
PCIe2: Bus 02 - 02
Net: eTSEC1, eTSEC3
Hit Ctrl-L to stop autoboot: 0
WARN: user forced bootcmd="run sysboot"
.. WARNING: adjusting available memory to 30000000
## Booting kernel from Legacy Image at 01000000 ...
```

```
Image Name: Linux-2.6.27.47
Created: 2015-11-20 10:22:39 UTC
Image Type: PowerPC Linux Kernel Image (gzip compressed)
Data Size: 8936305 Bytes = 8.5 MiB
Load Address: 00000000
Entry Point: 00000000
Verifying Checksum ... OK
## Flattened Device Tree blob at 00c00000
Booting using the fdt blob at 0x00c00000
Uncompressing Kernel Image ... OK
Loading Device Tree to 03ffb000, end 03fffe82 ... OK
setup_arch: bootmem
mpc85xx_fex_setup_arch()
arch: exit
[0.436112] Host controller irq 17
[0.477490] pci 0000:00:00.0: ignoring class b20 (doesn't match header type 01)
[0.566841] Assign root port irq 17 for 0000:00:00.0
[2.210329] Enabling all PCI devices
[2.802226] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[2.975494] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[3.889037]
[3.889041] Watchdog init<0>
Mount failed for selinuxfs on /selinux: No such file or directory
INIT: version 2.86 booting
Setting system clock: [ OK ]
Mounting all filesystems: [ OK ]
/sbin/dhclient-script: configuration for eth1 not found. Continuing with defaults.
/etc/sysconfig/network-scripts/network-functions: line 78: eth1: No such file or directory
Mounting system image: [ OK ]
Unpacking system image: [ OK ]
Uncompressing system image: [ OK ]
Loading system image: [ OK ]
net.ipv4.ip_forward = 0
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_no_pmtu_disc = 1
Starting internet superserver: inetd [ OK ]
net.core.rmem_max = 524288
net.core.wmem_max = 524288
net.core.rmem_default = 524288
net.core.wmem_default = 524288
net.core.somaxconn = 1024
net.core.netdev_max_backlog = 1024
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[23.255118] Device eth0 configured with sgmii interface
Non issu restart
[24.151321]
[24.151327] base_addr is 26524<0>
Secure erase requested! Please, do not power off module!
Starting the secure erase. !!
This may take time. Please wait !!
>>>> Wiping all storage devices ...
[28.706882] NX-OS starts punching watchdog
grep: Backu: No such file or directory
+++ Starting mtd secure erase for the partition /dev/mtd2 +++
Erasing /dev/mtd2 ...
Erasing 128 Kibyte @ 17e0000 -- 99 % complete.
```

```
---> SUCCESS
Writing random data onto /dev/mtd2
Filling /dev/mtd2 using random data ...
Erasing blocks: 192/192 (100%)
Writing data: 24576k/24576k (100%)
Verifying data: 24576k/24576k (100%)
---> SUCCESS
Erasing /dev/mtd2 ...
Erasing 128 Kibyte @ 17e0000 -- 99 % complete.
---> SUCCESS
+++ Skipping cmos secure erase +++
>>>> Done
+++ Skipping nvram secure erase +++
>>>> Done
>>>> Iniatilzing system to factory defaults ...
+++ Starting init-system +++
Initializing /dev/mtd5
/isan/bin/mount_jffs2.sh: line 68: ${LOG_FILE}: ambiguous [ 651.954326] Restarting system.
U-Boot 2011.12 (Jun 25 2014 - 16:28:41) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Golden image
U-boot retry count 1
Jump to upgradeable image at 0xefd20040
U-Boot 2011.12 (Jun 25 2014 - 16:19:54) Cisco Systems
CPU0: P1020E, Version: 1.1, (0x80ec0011)
Core: E500, Version: 5.1, (0x80212051)
Clock Configuration:
CPU0:666.667 MHz, CPU1:666.667 MHz,
CCB:333.333 MHz,
DDR:333.333 MHz (666.667 MT/s data rate) (Asynchronous), LBC:83.333 MHz
L1: D-cache 32 kB enabled
I-cache 32 kB enabled
Board: P1020FEX
[MCPSUMR 0x00000000, RSTRSCR 0x00000000, AUTORSTSR 0x0000c000]
I2C buses: ready
Upgradeable image
DRAM: Configuring DDR for 666.667 MT/s data rate
Time-out count = 480
DDR configuration get done
1 GiB (DDR3, 32-bit, CL=6, ECC on)
Memory test from 0x40000 to 0x1fdfffff
Data line test..... OK
Address line test..... OK
OK
Flash: 288 MiB
L2: 256 KB enabled
Set dbglevel to its default value (0x1)
PCIe1: Root Complex of mini PCIe SLOT, x1, regs @ 0xffe0a000
PCIe1: Bus 00 - 01
PCIe2: Root Complex of PCIe SLOT, no link, regs @ 0xffe09000
PCIe2: Bus 02 - 02
Net: eTSEC1, eTSEC3
Hit Ctrl-L to stop autoboot: 0
WARN: user forced bootcmd="run sysboot"
```

```
.. WARNING: adjusting available memory to 30000000
## Booting kernel from Legacy Image at 01000000 ...
Image Name: Linux-2.6.27.47
Created: 2015-11-20 10:22:39 UTC
Image Type: PowerPC Linux Kernel Image (gzip compressed)
Data Size: 8936305 Bytes = 8.5 MiB
Load Address: 00000000
Entry Point: 00000000
Verifying Checksum ... OK
## Flattened Device Tree blob at 00c00000
Booting using the fdt blob at 0x00c00000
Uncompressing Kernel Image ... OK
Loading Device Tree to 03ffb000, end 03fffe82 ... OK
setup_arch: bootmem
mpc85xx_fex_setup_arch()
arch: exit
[ 0.436112] Host controller irq 17
[ 0.477490] pci 0000:00:00.0: ignoring class b20 (doesn't match header type 01)
[ 0.566841] Assign root port irq 17 for 0000:00:00.0
[ 2.210556] Enabling all PCI devices
[ 2.804559] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[ 2.975502] FSL:i2c-mpc - probing i2c controller
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[ 3.889014]
[ 3.889018] Watchdog init<0>
Mount failed for selinuxfs on /selinux: No such file or directory
INIT: version 2.86 booting
Setting system clock: [ OK ]
Mounting all filesystems: [ OK ]
/sbin/dhclient-script: configuration for eth1 not found. Continuing with defaults.
/etc/sysconfig/network-scripts/network-functions: line 78: eth1: No such file or directory
Mounting system image: [ OK ]
Unpacking system image: [ OK ]
Uncompressing system image: [ OK ]
Loading system image: [ OK ]
net.ipv4.ip_forward = 0
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_no_pmtu_disc = 1
Starting internet superserver: inetd [ OK ]
net.core.rmem_max = 524288
net.core.wmem_max = 524288
net.core.rmem_default = 524288
net.core.wmem_default = 524288
net.core.somaxconn = 1024
net.core.netdev_max_backlog = 1024
modprobe: FATAL: Could not load /lib/modules/2.6.27.47/modules.dep: No such file or directory
[ 22.630994] Device eth0 configured with sgmii interface
Non issu restart
[ 23.535827]
[ 23.535832] base_addr is 26524<0>
INIT: Entering runlevel: 3
fex login: Sorry, user root is not allowed to execute '/sbin/sysctl -q -w vm.drop_caches=3'
 as root on fex.
[ 28.090052] NX-OS starts punching watchdog
fex login:
```

The following is an example output for configuring secure erase factory reset command on module:

```
switch# factory-reset [all | module <mod>]
switch# factory-reset [module <3>]
!!!! WARNING !!!!
The factory reset operation will erase ALL persistent storage on the specified module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken to render data non-recoverable. Please, proceed with caution and
understanding that this operation cannot be undone and will leave the system in a
fresh-from-factory state.
!!!! WARNING !!!!
Continue? (y/n) [n] y
A module reload is required for the reset operation to proceed. Please, wait...
…truncated…
Secure erase requested! Please, do not power off module!
>>>> Wiping all storage devices ...
+++ Starting mmc secure erase for /dev/mmcblk0 +++
*** Please, wait - this may take several minutes ***
\
---> SUCCESS
+++ Starting SSD secure erase for /dev/sda +++
*** Please, wait - this may take several minutes ***
\
---> SUCCESS
+++ Starting cmos secure erase +++
\
---> SUCCESS
>>>> Done
+++ Starting nvram secure erase +++
\
---> SUCCESS
>>>> Done
```

The following is an example output logs for configuring secure erase factory reset command on LC:

```
switch#
switch# factory-reset mod 1
!!!! WARNING !!!!
The factory reset operation will erase ALL persistent storage on the specified module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable.
Please, proceed with
caution and understanding that this operation cannot be undone and will leave the system
in
a fresh-from-factory state.
!!!! WARNING !!!!
Continue? (y/n) [n] y
A module reload is required for the reset operation to proceed. Please, wait...
reloading module 1 ...
.....................................................................
SUCCESS! All persistent storage devices detected on the specified module have been purged.
switch#
```

The following is an example output logs for configuring secure erase factory reset command on mod:

```
switch# factory-reset mod 26
!!!! WARNING !!!!
The factory reset operation will erase ALL persistent storage on the specified module.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable.
Please, proceed with
caution and understanding that this operation cannot be undone and will leave the system
in
a fresh-from-factory state.
!!!! WARNING !!!!
Continue? (y/n) [n] y
A module reload is required for the reset operation to proceed. Please, wait...
```

```
reloading module 26 ...
.....................................................................
.........................
.....................................................................
.........................
.....................................................................
.........................
.....................................................................
.........................
.....................................................................
.........................
.....................................................................
.........................
.....................................................................
.........................
.....................................................
SUCCESS! All persistent storage devices detected on the specified module have been cleared.
>>>> Please, note - multiple write passes were required to remove data from one or more
devices. <<<<
switch#
```

Configuring Secure Erase



INDEX

A

access restriction, configuring 43
  ntp 43
action statements 147
  EEM 147
action statements, configuring 154
  EEM 154
activating sessions 188
  SPAN 188
Active Buffer Monitoring 229–230
  configuring 230
  overview 229
adding show commands, alert groups 88
  smart call home 88
alert groups 75
  smart call home 75
associating alert groups 87
  smart call home 87

B

buffer histogram data 230–231
  accessing 230
  buffer histogram data 230
    collecting 230
  displaying 231
buffer monitoring 230
  configuring 230

C

call home notifications 95
  full-text format for syslog 95
  XML format for syslog 95
clock manager 38
  ntp 38
commiting changes 48
  ntp configuration 48
configuration example 219–220
  ERSPAN 219–220
    destination 220
    source 219
configuration examples 51
  ntp 51
configuration, verifying 112
  scheduler 112
contact information, configuring 83
  smart call home 83
creating, deleting sessions 184
  SPAN 184

D

default ntp settings 40
default parameters 205
  ERSPAN 205
default settings 81, 102, 104, 149
  EEM 149
  rollback 102
  scheduler 104
  smart call home 81
default SNMP settings 119
defining EEM policies 156
  VSH script 156
description, configuring 187
  SPAN 187
destination ports, characteristics 181
  SPAN 181
destination profile, creating 85
  smart call home 85
destination profile, modifying 86
  smart call home 86
destination profiles 74
  smart call home 74
destinations 180
  SPAN 180
device IDs 77
  call home format 77
diagnostics 139–142
  configuring 141
  default settings 142
  expansion modules 141
  health monitoring 140
  runtime 139
disabling 69, 112
  DOM logging 69
  scheduler 112
displaying information 193
  SPAN 193

displaying installation log information 244
duplicate message throttling, disabling 92
smart call home 92

E

e-mail details, configuring 89
smart call home 89
e-mail notifications 73
smart call home 73
EEM 146–150, 152, 154, 157–159
action statements 147
action statements, configuring 154
default settings 149
defining environment variables 149
event statements 146
event statements, configuring 152
policies 146
prerequisites 148
syslog script 159
system policies,overriding 158
user policy, defining 150
VSH script 157
registering and activating 157
VSH script policies 148
embedded event manager 145
overview 145
enabling 68, 105
DOM logging 68
scheduler 105
environment variables, defining 149
EEM 149
ERSPAN 201–203, 205–206, 209, 219–220
configuring destination sessions 209
configuring source sessions 206
default parameters 205
destination 220
configuration example 220
destination sessions 209
configuring for ERSPAN 209
destinations 202
high availability 203
information about 201
prerequisites 203
related documents 220
sessions 203
multiple 203
source 219
configuration example 219
source sessions 206
configuring for ERSPAN 206
sources 201
types 201
Ethernet destination port, configuring 184
SPAN 184
event statements 146
EEM 146
event statements, configuring 152
EEM 152
example 113
job schedule, displaying 113
scheduler job, creating 113
scheduler job, scheduling 113
scheduler jobs, displaying results 113
executing a session 101

F

facility messages logging 59
configuring 59
feature groups, creating 284
RBAC 284
feature history 52
ntp 52
filtering SNMP requests 122

G

GOLD diagnostics 139–141
configuring 141
expansion modules 141
health monitoring 140
runtime 139
guidelines 38
ntp 38
guidelines and limitations 54, 81, 104, 119, 280
scheduler 104
smart call home 81
SNMP 119
system message logging 54
user accounts 280
guidelines and limitations for configuration rollback 263

H

health monitoring diagnostics 140
information 140
high availability 18
PTP 18
high availability 18

I

IDs 77
serial IDs 77
information 37
ntp 37
information about 103
scheduler 103

interfaces, configuring 22
PTP 22

J

job schedule, displaying 113
example 113
job, deleting 108
scheduler 108

L

linkDown notifications 128–129
linkUp notifications 128–129
log file size, defining 105
scheduler 105
log file, clearing 111
scheduler 111
log files 104
scheduler 104
logging 59
facility messages 59
module messages 59

M

message encryption 121
SNMP 121
module messages logging 59
configuring 59

N

notification receivers 123
SNMP 123
ntp 37–38, 40, 43, 51–52
access restriction, configuring 43
clock manager 38
configuration examples 51
default settings 40
feature history 52
guidelines 38
information 37
related documents 52
time server 38
using cfs 38
virtualization 38
ntp configuration 48
commiting changes 48
ntp using cfs 38

O

overview 145
embedded event manager 145

P

password requirements 279
periodic inventory notifications, configuring 91
smart call home 91
policies 146
EEM 146
prerequisites 148, 203
EEM 148
ERSPAN 203
PTP 15–17, 19–20, 22
configuring globally 20
default settings 19
device types 16
interface, configuring 22
overview 15
process 17

R

RBAC 277–279, 281–282, 284–286
feature groups, creating 284
rules 278
user account restrictions 279
user accounts, configuring 281
user role interface policies, changing 284
user role VLAN policies, changing 285
user roles 277
user roles and rules, configuring 282
verifying 286
registering 82
smart call home 82
related documents 52, 220
ERSPAN 220
ntp 52
remote user authentication 104
scheduler 104
remote user authentication, configuring 106–107
scheduler 106–107
requirements 279
user passwords 279
roles 277
authentication 277
rollback 99, 102
checkpoint copy 99
creating a checkpoint copy 99
default settings 102
deleting a checkpoint file 99
description 99
example configuration 99
guidelines 99
high availability 99
implementing a rollback 99
limitations 99
reverting to checkpoint file 99
verifying configuration 102

rules 278
RBAC 278
runtime diagnostics 139
information 139

S

scheduler 103–109, 111–112, 114
configuration, verifying 112
default settings 104
disabling 112
enabling 105
guidelines and limitations 104
information about 103
job, deleting 108
log file size, defining 105
log file, clearing 111
log files 104
remote user authentication 104
remote user authentication, configuring 106–107
standards 114
timetable, defining 109
scheduler job, creating 113
example 113
scheduler job, scheduling 113
example 113
scheduler jobs, displaying results 113
example 113
serial IDs 77
description 77
server IDs 77
description 77
session manager 99, 101–102
committing a session 101
configuring an ACL session (example) 102
description 99
discarding a session 101
guidelines 99
limitations 99
saving a session 101
verifying configuration 102
verifying the session 101
smart call home 73–75, 81–83, 85–89, 91–94
adding show commands, alert groups 88
alert groups 75
associating alert groups 87
contact information, configuring 83
default settings 81
description 73
destination profile, creating 85
destination profile, modifying 86
destination profiles 74
duplicate message throttling, disabling 92
e-mail details, configuring 89
guidelines and limitations 81
smart call home *(continued)*
message format options 74
periodic inventory notifications 91
prerequisites 81
registering 82
testing the configuration 93
verifying 94
smart call home messages 74, 76
configuring levels 76
format options 74
SMUs 237–239, 241, 243–244
activating packages 241
adding packages 241
committing the active package set 243
deactivating packages 243
described 237
guidelines 239
limitations 239
package management 238
preparing for package installation 239
prerequisites 238
removing packages 243
SNMP 115–123, 125, 131
access groups 119
configuring users 120
default settings 119
disabling 131
filtering requests 122
functional overview 115
group-based access 119
guidelines and limitations 119
inband access 125
message encryption 121
notification receivers 123
security model 117
trap notifications 116
user synchronization with CLI 118
user-based security 117
SNMP 117
version 3 security features 116
SNMP (Simple Network Management Protocol) 116
versions 116
SNMP notification receivers 124
configuring with VRFs 124
SNMP notifications 124
filtering based on a VRF 124
snmp-server name 120
SNMPv3 116, 121
assigning multiple roles 121
security features 116
source IDs 77
call home event format 77
source ports, characteristics 180
SPAN 180
source ports, configuring 186
SPAN 186

SPAN 179–181, 184, 186–188, 193
activating sessions 188
characteristics, source ports 180
creating, deleting sessions 184
description, configuring 187
destination ports, characteristics 181
destinations 180
displaying information 193
egress sources 180
Ethernet destination port, configuring 184
ingress sources 180
source port channels, configuring 186
source ports, configuring 186
sources for monitoring 179
VLANs, configuring 186
SPAN sources 180
egress 180
ingress 180
standards 114
scheduler 114
Switched Port Analyzer 179
syslog 63, 159
configuring 63
EEM 159
system message logging 53–54
guidelines and limitations 54
information about 53
system message logging settings 55
defaults 55
system policies, overriding 158
EEM 158

T

testing the configuration 93
smart call home 93
time server 38
ntp 38
timetable, defining 109
scheduler 109
trap notifications 116

U

user account restrictions 279
RBAC 279
user accounts 279–280, 286
guidelines and limitations 280
passwords 279
verifying 286
user policies, defining 150
EEM 150
user role interface policies, changing 284
RBAC 284
user role VLAN policies, changing 285
RBAC 285
user roles 277
RBAC 277
user roles and rules, creating 282
RBAC 282
users 277
description 277

V

verifying 69, 94, 286
DOM logging configuration 69
RBAC 286
smart call home 94
user accounts 286
virtualization 38
ntp 38
VRFs 124
configuring SNMP notification receivers with 124
filtering SNMP notifications 124
VSH script 156
defining EEM policies 156
VSH script policies 148, 157
EEM 148
registering and activating 157

W

warp mode 225–227
disabling 226
enabling 226
overview 225
verifying the status of 227