



Cisco Nexus 3600 Switch NX-OS System Management Configuration Guide, Release 10.4(x)

First Published: 2023-08-18

Last Modified: 2024-03-29

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: <http://www.cisco.com/go/softwareterms>. Cisco product warranty information is available at <http://www.cisco.com/go/warranty>. US Federal Communications Commission Notices are found here <http://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: [www.cisco.com go trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2023–2024 Cisco Systems, Inc. All rights reserved.



CONTENTS

Trademarks ?

PREFACE

Preface xv

Audience xv

Document Conventions xv

Related Documentation for Cisco Nexus 3600 Platform 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 4

Supported Platforms 4

CHAPTER 3

Two-stage Configuration Commit 5

About Two-stage Configuration Commit 5

Guidelines and Limitations 6

Configuring in Two-Stage Configuration Commit Mode 6

Aborting the Two-Stage Configuration Commit Mode 10

Displaying Commit IDs 10

Rollback Capability 11

Viewing Current Session Configurations 11

CHAPTER 4	Configuring Switch Profiles	13
	About Switch Profiles	13
	Switch Profile Configuration Modes	14
	Configuration Validation	15
	Software Upgrades and Downgrades with Switch Profiles	16
	Prerequisites for Switch Profiles	16
	Guidelines and Limitations for Switch Profiles	16
	Configuring Switch Profiles	17
	Adding a Switch to a Switch Profile	19
	Adding or Modifying Switch Profile Commands	21
	Importing a Switch Profile	23
	Verifying Commands in a Switch Profile	25
	Isolating a Peer Switch	26
	Deleting a Switch Profile	27
	Deleting a Switch from a Switch Profile	28
	Displaying the Switch Profile Buffer	29
	Synchronizing Configurations After a Switch Reboot	30
	Switch Profile Configuration show Commands	30
	Supported Switch Profile Commands	31
	Configuration Examples for Switch Profiles	32
	Creating a Switch Profile on a Local and Peer Switch Example	32
	Verifying the Synchronization Status Example	33
	Displaying the Running Configuration	34
	Displaying the Switch Profile Synchronization Between Local and Peer Switches	34
	Displaying Verify and Commit on Local and Peer Switches	35
	Successful and Unsuccessful Synchronization Examples	36
	Configuring the Switch Profile Buffer, Moving the Buffer, and Deleting the Buffer	36
CHAPTER 5	Configuring PTP	39
	About PTP	39
	PTP Device Types	39
	PTP Process	40
	High Availability for PTP	41

Guidelines and Limitations for PTP	41
Default Settings for PTP	41
Configuring PTP	42
Configuring PTP Globally	42
Configuring PTP on an Interface	44
Verifying the PTP Configuration	46

CHAPTER 6
Configuring NTP 47

Information About NTP	47
NTP as Time Server	48
Distributing NTP Using CFS	48
Clock Manager	48
High Availability	48
Virtualization Support	48
Prerequisites for NTP	49
Guidelines and Limitations for NTP	49
Default Settings	50
Configuring NTP	50
Enabling or Disabling NTP on an Interface	50
Configuring the Device as an Authoritative NTP Server	51
Configuring an NTP Server and Peer	52
Configuring NTP Authentication	53
Configuring NTP Access Restrictions	55
Configuring the NTP Source IP Address	56
Configuring the NTP Source Interface	57
Configuring an NTP Broadcast Server	58
Configuring an NTP Multicast Server	59
Configuring an NTP Multicast Client	60
Configuring NTP Logging	60
Enabling CFS Distribution for NTP	61
Committing NTP Configuration Changes	62
Discarding NTP Configuration Changes	62
Releasing the CFS Session Lock	63
Verifying the NTP Configuration	63

Configuration Examples for NTP 64

CHAPTER 7**Configuring Session Manager 67**

About Session Manager 67

Guidelines and Limitations for Session Manager 67

Configuring Session Manager 68

Creating a Session 68

Configuring ACLs in a Session 68

Verifying a Session 69

Committing a Session 69

Saving a Session 69

Discarding a Session 69

Configuration Example for Session Manager 69

Verifying the Session Manager Configuration 70

CHAPTER 8**Configuring Smart Call Home 71**

About Smart Call Home 71

Smart Call Home Overview 72

Smart Call Home Destination Profiles 72

Smart Call Home Alert Groups 73

Smart Call Home Message Levels 74

Call Home Message Formats 75

Guidelines and Limitations for Smart Call Home 79

Prerequisites for Smart Call Home 79

Default Call Home Settings 80

Configuring Smart Call Home 80

Registering for Smart Call Home 80

Configuring Contact Information 81

Creating a Destination Profile 83

Modifying a Destination Profile 84

Associating an Alert Group with a Destination Profile 85

Adding Show Commands to an Alert Group 86

Configuring E-Mail Server Details 87

Configuring Periodic Inventory Notifications 88

Disabling Duplicate Message Throttling	89
Enabling or Disabling Smart Call Home	90
Testing the Smart Call Home Configuration	91
Verifying the Smart Call Home Configuration	92
Sample Syslog Alert Notification in Full-Text Format	92
Sample Syslog Alert Notification in XML Format	93

CHAPTER 9**Configuring the Scheduler 97**

Information About the Scheduler	97
Remote User Authentication	98
Scheduler Log Files	98
Guidelines and Limitations for the Scheduler	98
Default Settings for the Scheduler	98
Configuring the Scheduler	99
Enabling the Scheduler	99
Defining the Scheduler Log File Size	99
Configuring Remote User Authentication	100
Defining a Job	101
Deleting a Job	102
Defining a Timetable	103
Clearing the Scheduler Log File	105
Disabling the Scheduler	106
Verifying the Scheduler Configuration	106
Configuration Examples for the Scheduler	107
Creating a Scheduler Job	107
Scheduling a Scheduler Job	107
Displaying the Job Schedule	107
Displaying the Results of Running Scheduler Jobs	107
Standards for the Scheduler	108

CHAPTER 10**Configuring SNMP 109**

About SNMP	109
SNMP Functional Overview	109
SNMP Notifications	110

- SNMPv3 110
 - Security Models and Levels for SNMPv1, v2, and v3 110
 - User-Based Security Model 111
 - CLI and SNMP User Synchronization 112
 - Group-Based SNMP Access 113
- Guidelines and Limitations for SNMP 113
- Default SNMP Settings 114
- Configuring SNMP 114
 - Configuring the SNMP Source Interface 114
 - Configuring SNMP Users 115
 - Generating Hashed Password Offline 117
 - Enforcing SNMP Message Encryption 117
 - Assigning SNMPv3 Users to Multiple Roles 118
 - Creating SNMP Communities 118
 - Filtering SNMP Requests 118
 - Configuring SNMP Notification Receivers 119
 - Configuring SNMP Notification Receivers with VRFs 120
 - Filtering SNMP Notifications Based on a VRF 121
 - Configuring SNMP for Inband Access 121
 - Enabling SNMP Notifications 123
 - Configuring Link Notifications 125
 - Disabling Link Notifications on an Interface 125
 - Enabling One-Time Authentication for SNMP over TCP 126
 - Assigning SNMP Switch Contact and Location Information 126
 - Configuring the Context to Network Entity Mapping 127
 - Configuring the SNMP Local Engine ID 127
 - Disabling SNMP 128
 - Verifying the SNMP Configuration 129

CHAPTER 11 **Using the PCAP SNMP Parser 131**

- Using the PCAP SNMP Parser 131

CHAPTER 12 **Configuring RMON 133**

- Information About RMON 133

RMON Alarms	133
RMON Events	134
Configuration Guidelines and Limitations for RMON	134
Verifying the RMON Configuration	134
Default RMON Settings	135
Configuring RMON Alarms	135
Configuring RMON Events	136

CHAPTER 13

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

CHAPTER 14

Configuring the Embedded Event Manager	143
About Embedded Event Manager	143
Embedded Event Manager Policies	143
Event Statements	144
Action Statements	145
VSH Script Policies	146
Licensing Requirements for Embedded Event Manager	146
Prerequisites for Embedded Event Manager	146
Guidelines and Limitations for Embedded Event Manager	146
Default Settings for Embedded Event Manager	147
Configuring Embedded Event Manager	147
Defining an Environment Variable	147
Defining a User Policy Using the CLI	148
Configuring Event Statements	149
Configuring Action Statements	152
Defining a Policy Using a VSH Script	154

Registering and Activating a VSH Script Policy	154
Overriding a System Policy	155
Configuring Syslog as an EEM Publisher	157
Verifying the Embedded Event Manager Configuration	158
Event Log Auto-Collection and Backup	158
Extended Log File Retention	159
Trigger-Based Event Log Auto-Collection	163
Local Log File Storage	171
External Log File Storage	173
Verifying the Embedded Event Manager Configuration	174
Configuration Examples for Embedded Event Manager	175
Additional References	176

CHAPTER 15	Configuring Onboard Failure Logging	177
	About OBFL	177
	Prerequisites for OBFL	178
	Guidelines and Limitations for OBFL	178
	Default Settings for OBFL	178
	Configuring OBFL	178
	Verifying the OBFL Configuration	181
	Configuration Example for OBFL	182
	Additional References	182
	Related Documents	182

CHAPTER 16	Configuring SPAN	183
	Information About SPAN	183
	SPAN Sources	183
	Characteristics of Source Ports	184
	SPAN Destinations	184
	Characteristics of Destination Ports	184
	Guidelines and Limitations for SPAN	185
	Creating or Deleting a SPAN Session	186
	Configuring an Ethernet Destination Port	186
	Configuring Source Ports	188

Configuring the Rate Limit for SPAN Traffic	188
Configuring Source Port Channels or VLANs	189
Configuring the Description of a SPAN Session	190
Activating a SPAN Session	191
Suspending a SPAN Session	191
Displaying SPAN Information	192
Configuration Examples for SPAN	193
Configuration Example for a SPAN Session	193
Configuration Example for a Unidirectional SPAN Session	193
Configuration Example for a SPAN ACL	194
Configuration Examples for UDF-Based SPAN	195

CHAPTER 17
Configuring ERSPAN 197

About ERSPAN	197
ERSPAN Sources	197
Multiple ERSPAN Sessions	198
High Availability	198
Prerequisites for ERSPAN	198
Guidelines and Limitations for ERSPAN	198
Default Settings for ERSPAN	201
Configuring ERSPAN	201
Configuring an ERSPAN Source Session	201
Configuring SPAN Forward Drop Traffic for ERSPAN Source Session	204
Configuring an ERSPAN ACL	206
Configuring User Defined Field (UDF) Based ACL Support	207
Configuring IPv6 User Defined Field (UDF) on ERSPAN	209
Shutting Down or Activating an ERSPAN Session	212
Verifying the ERSPAN Configuration	214
Configuration Examples for ERSPAN	214
Configuration Example for an ERSPAN Source Session	214
Configuration Example for an ERSPAN ACL	214
Configuration Examples for UDF-Based ERSPAN	215
Additional References	216
Related Documents	216

CHAPTER 18**Configuring DNS 217**

- About DNS Client 217
 - Name Servers 217
 - DNS Operation 217
 - High Availability 218
- Prerequisites for DNS Clients 218
- Default Settings for DNS Clients 218
- Configuring the DNS Source Interface 218
- Configuring DNS Clients 219

CHAPTER 19**Configuring sFlow 223**

- About sFlow 223
 - sFlow Agent 223
- Prerequisites 224
- Guidelines and Limitations for sFlow 224
- Default Settings for sFlow 224
- Minimum Requirements for Sampling 224
- Configuring sFlow 225
 - Enabling the sFlow Feature 225
 - Configuring the Sampling Rate 225
 - Configuring the Maximum Sampled Size 226
 - Configuring the Counter Poll Interval 227
 - Configuring the Maximum Datagram Size 228
 - Configuring the sFlow Analyzer Address 229
 - Configuring the sFlow Analyzer Port 230
 - Configuring the sFlow Agent Address 230
 - Configuring the sFlow Sampling Data Source 231
- Verifying the sFlow Configuration 232
- Configuration Examples for sFlow 233
- Additional References for sFlow 233

CHAPTER 20**Configuring Graceful Insertion and Removal 235**

- About Graceful Insertion and Removal 235

Profiles	236
Snapshots	237
GIR Workflow	237
Configuring the Maintenance-Mode Profile	238
Configuring the Normal-Mode Profile	239
Creating a Snapshot	240
Adding Show Commands to Snapshots	242
Triggering Graceful Removal	243
Triggering Graceful Insertion	246
Maintenance Mode Enhancements	247
Verifying the GIR Configuration	248

CHAPTER 21

Performing Configuration Replace	251
About Configuration Replace and Commit-timeout	251
Overview	251
Benefits of Configuration Replace	253
Guidelines and Limitations for Configuration Replace	253
Recommended Workflow for Configuration Replace	255
Performing a Configuration Replace	256
Verifying Configuration Replace	258
Examples for Configuration Replace	258

CHAPTER 22

Configuring Rollback	265
About Rollbacks	265
Guidelines and Limitations for Rollbacks	265
Creating a Checkpoint	266
Implementing a Rollback	267
Verifying the Rollback Configuration	268

CHAPTER 23

Integrity Check of Candidate Config	269
About Candidate Config	269
Guidelines and Limitations for Candidate Config Integrity Check	269
Performing Integrity Check for Candidate Config	274
Examples of Integrity Check	274

CHAPTER 24	Configuring User Accounts and RBAC	277
	About User Accounts and RBAC	277
	User Roles	277
	Rules	278
	User Role Policies	278
	User Account Configuration Restrictions	278
	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
	Default Settings for the User Accounts and RBAC	286



Preface

This preface includes the following sections:

- [Audience, on page xv](#)
- [Document Conventions, on page xv](#)
- [Related Documentation for Cisco Nexus 3600 Platform Switches, on page xvi](#)
- [Documentation Feedback, on page xvi](#)
- [Communications, Services, and Additional Information, on page xvi](#)

Audience

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

Document Conventions

Command descriptions use the following conventions:

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

Convention	Description
<i>variable</i>	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
<code>screen font</code>	Terminal sessions and information the switch displays are in screen font.
boldface screen font	Information you must enter is in boldface screen font.
<i>italic screen font</i>	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 3600 Platform Switches

The entire Cisco Nexus 3600 platform switch documentation set is available at the following URL:

<http://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 impact you're looking for with the technologies that matter, visit [Cisco Services](#).
- To submit a service request, visit [Cisco Support](#).
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit [Cisco Marketplace](#).
- 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 3600 Series NX-OS System Management Configuration Guide, Release 10.4(x)* and where they are documented.

Table 1: New and Changed Features

Feature	Description	Changed in Release	Where Documented
Unified Configuration Difference	On the Cisco Nexus switches, the show diff running-config command provides the merged option to merge the sub-command details instead of replacing.	10.4(3)F	Guidelines and Limitations for Candidate Config Integrity Check, on page 269 Performing Integrity Check for Candidate Config, on page 274 Examples of Integrity Check, on page 274
Partial diff support for polymorphic commands	Added partial diff support for polymorphic commands	10.4(3)F	Guidelines and Limitations for Candidate Config Integrity Check, on page 269
CR support for polymorphic commands	Added CR support for polymorphic commands	10.4(3)F	Guidelines and Limitations for Configuration Replace, on page 253
CR multiline support	Added configuration replace feature support for LDAP.	10.4(2)F	Guidelines and Limitations for Configuration Replace, on page 253
NA	No new features added for this release.	10.4(1)F	NA



CHAPTER 2

Overview

This chapter contains the following sections:

- [System Management Features, on page 3](#)
- [Licensing Requirements, on page 4](#)
- [Supported Platforms, on page 4](#)

System Management Features

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

Feature	Description
User Accounts and RBAC	User accounts and role-based access control (RBAC) allow you to define the rules for an assigned role. Roles restrict the authorization that the user has to access management operations. Each user role can contain multiple rules and each user can have multiple roles.
Session Manager	Session Manager allows you to create a configuration and apply it in batch mode after the configuration is reviewed and verified for accuracy and completeness.
Online Diagnostics	Cisco Generic Online Diagnostics (GOLD) define a common framework for diagnostic operations across Cisco platforms. The online diagnostic framework specifies the platform-independent fault-detection architecture for centralized and distributed systems, including the common diagnostics CLI and the platform-independent fault-detection procedures for boot-up and run-time diagnostics. The platform-specific diagnostics provide hardware-specific fault-detection tests and allow you to take appropriate corrective action in response to diagnostic test results.

Feature	Description
Configuration Rollback	The configuration rollback feature allows users to take a snapshot, or user checkpoint, of the Cisco NX-OS configuration and then reapply that configuration to a switch at any point without having to reload the switch. A rollback allows any authorized administrator to apply this checkpoint configuration without requiring expert knowledge of the features configured in the checkpoint.
SNMP	The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.
RMON	RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. Cisco NX-OS supports RMON alarms, events, and logs to monitor Cisco NX-OS devices.
SPAN	The Switched Port Analyzer (SPAN) feature (sometimes called port mirroring or port monitoring) selects network traffic for analysis by a network analyzer. The network analyzer can be a Cisco SwitchProbe or other Remote Monitoring (RMON) probes.

Licensing Requirements

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

Supported Platforms

Starting with Cisco NX-OS release 7.0(3)I7(1), use the [Nexus Switch Platform Support Matrix](#) to know from which Cisco NX-OS releases various Cisco Nexus 9000 and 3000 switches support a selected feature.



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:

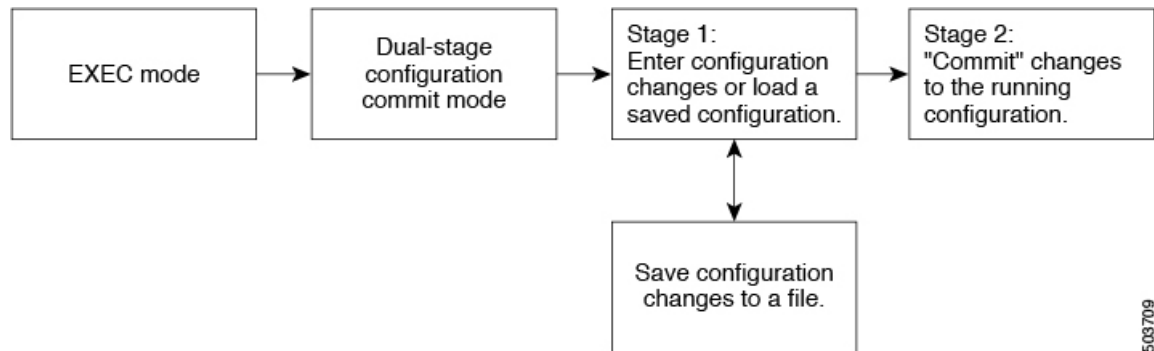
- [About Two-stage Configuration Commit, on page 5](#)
- [Guidelines and Limitations, on page 6](#)
- [Configuring in Two-Stage Configuration Commit Mode, on page 6](#)
- [Aborting the Two-Stage Configuration Commit Mode, on page 10](#)
- [Displaying Commit IDs, on page 10](#)
- [Rollback Capability, on page 11](#)
- [Viewing Current Session Configurations, on page 11](#)

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



503709

Guidelines and Limitations

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

- This feature is supported only for a CLI interface in a user-interactive session.
- Before you run any feature-related configuration commands, enable the feature using the **feature** command and commit it using the **commit** command.
- Two-stage configuration commit mode doesn't support other modes like maintenance mode, scheduler mode, or virtual mode.
- When you're in the two-stage configuration commit mode, avoid editing configurations in one-stage configuration commit mode from different sessions at the same time.
- Review the configurations using the **show configuration** command before committing the changes.
- 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	<p>configure dual-stage</p> <p>Example:</p> <pre>switch# configure dual-stage switch(config-dual-stage)#</pre>	<p>Creates a new target configuration session.</p> <p>Note The target configuration isn't a copy of the running configuration. It has only the configuration commands entered during the target configuration session.</p>
Step 2	<p>feature <i>feature_name</i></p> <p>Example:</p> <pre>switch(config-dual-stage)# feature bgp switch(config-dual-stage)#</pre>	<p>Enables the feature.</p> <p>Note</p> <ul style="list-style-type: none"> 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	<p>commit [<i>confirmedseconds</i>]</p> <p>Example:</p> <pre>switch(config-dual-stage-router)# commit confirmed 30 Verification Succeeded. Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time. Configuration committed by user 'admin' using Commit ID : 1000000001 switch(config-dual-stage)# switch(config-dual-stage)# commit Confirming commit for trial session. switch(config-dual-stage)#</pre> <p>Example:</p> <pre>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)#</pre>	<p>Commits changes to the running configuration.</p> <ul style="list-style-type: none"> 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. <p>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.</p>
Step 4	<p>Example:</p> <pre>switch(config-dual-stage)# router bgp 64515.46 switch(config-dual-stage-router)# switch(config-dual-stage-router)# router-id</pre>	<p>Run any feature-related commands that are supported in this configuration mode.</p>

	Command or Action	Purpose
	141.8.139.131 switch(config-dual-stage-router)#	
Step 5	<p>show configuration</p> <p>Example:</p> <pre>switch(config-dual-stage-router)# show configuration ! Cached configuration ! router bgp 64515.46 router-id 141.8.139.131</pre>	<p>Displays the target configuration.</p> <p>Note You can run this command only in the dual-stage configuration mode.</p>
Step 6	<p>commit [confirmed <i>seconds</i>]</p> <p>Example:</p> <pre>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</pre>	Commits changes to the running configuration.
Step 7	<p>(Optional) show configuration commit [changes] <i>commit-id</i></p> <p>Example:</p> <pre>switch(config-dual-stage-router)# show configuration commit changes 1000000003 *** /bootflash/.dual-stage/1000000003.tmp Fri Mar 19 10:59:00 2021 --- /bootflash/.dual-stage/1000000003 Fri Mar 19 10:59:05 2021 ***** *** 378,383 **** --- 378,385 ---- 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</pre> <p>Example:</p> <pre>switch(config-dual-stage)# show configuration commit 1000000003 feature bgp router bgp 64515.46 router-id 141.8.139.131 . . .</pre>	<p>Displays commit-related information.</p> <p>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.</p> <p>Use the show configuration commit changes <i>commit-id</i> command to view only the changes in the current session of the commit you specify.</p> <p>Use the show configuration commit <i>commit-id</i> command to view the complete configurations in the commit you specify.</p>

	Command or Action	Purpose
Step 8	<p>(Optional) save configuration <i>filename</i></p> <p>Example:</p> <pre>switch(config-dual-stage)# save configuration bootflash:test.cfg</pre>	<p>Saves the target configurations to a separate file without committing them to the running configuration.</p> <p>Note</p> <ul style="list-style-type: none"> • 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 <i>filename</i> command. • Some of the user-specific information will be masked based on the user role.
Step 9	<p>(Optional) load <i>filename</i></p> <p>Example:</p> <pre>switch (config-dual-stage)# show configuration ! Cached configuration switch (config-dual-stage)# load test.cfg switch (config-dual-stage-router)# show configuration ! Cached configuration ! router bgp 1 switch(config-dual-stage-router)#</pre>	<p>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 <i>filename</i> command.</p> <p>You can load a target configuration that you saved using only the save configuration <i>filename</i> command.</p>
Step 10	<p>(Optional) clear configuration</p> <p>Example:</p> <pre>switch(config-dual-stage)# show configuration ! Cached configuration ! router bgp 64515.46 router-id 141.8.139.131 switch (config-dual-stage)# clear configuration switch (config-dual-stage)# show configuration ! Cached configuration switch (config-dual-stage)#</pre>	<p>Clears changes made to the target configuration without terminating the configuration session. It deletes any configuration changes that aren't committed.</p>
Step 11	<p>end</p> <p>Example:</p> <pre>switch(config-dual-stage-if)# end Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]</pre>	<p>Exits the global dual stage configuration mode.</p> <p>If you end a configuration session without committing the configuration changes, you'll be prompted to save changes, discard changes, or cancel the action:</p> <ul style="list-style-type: none"> • 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
		<p>Note</p> <ul style="list-style-type: none"> • 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-ipmcmis
class-map type control-plane match-any copp-s-l2switched
class-map type control-plane match-any copp-s-l3destmiss
switch(config-dual-stage-cmap)#
```

If there is no configuration, the following message appears:

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

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



CHAPTER 4

Configuring Switch Profiles

This chapter contains the following sections:

- [About Switch Profiles, on page 13](#)
- [Switch Profile Configuration Modes, on page 14](#)
- [Configuration Validation, on page 15](#)
- [Software Upgrades and Downgrades with Switch Profiles, on page 16](#)
- [Prerequisites for Switch Profiles, on page 16](#)
- [Guidelines and Limitations for Switch Profiles, on page 16](#)
- [Configuring Switch Profiles, on page 17](#)
- [Adding a Switch to a Switch Profile, on page 19](#)
- [Adding or Modifying Switch Profile Commands, on page 21](#)
- [Importing a Switch Profile, on page 23](#)
- [Verifying Commands in a Switch Profile, on page 25](#)
- [Isolating a Peer Switch, on page 26](#)
- [Deleting a Switch Profile, on page 27](#)
- [Deleting a Switch from a Switch Profile, on page 28](#)
- [Displaying the Switch Profile Buffer, on page 29](#)
- [Synchronizing Configurations After a Switch Reboot, on page 30](#)
- [Switch Profile Configuration show Commands, on page 30](#)
- [Supported Switch Profile Commands, on page 31](#)
- [Configuration Examples for Switch Profiles, on page 32](#)

About Switch Profiles

Several applications require consistent configuration across Cisco Nexus Series switches. For example, with a Virtual Port Channel (vPC), you must have identical configurations. Mismatched configurations can cause errors or misconfigurations that can result in service disruptions.

The configuration synchronization (config-sync) feature allows you to configure one switch profile and have the configuration be automatically synchronized to the peer switch. A switch profile provides the following benefits:

- Allows configurations to be synchronized between switches.
- Merges configurations when connectivity is established between two switches.

- Provides control of exactly which configuration gets synchronized.
- Ensures configuration consistency across peers through merge and mutual-exclusion checks.
- Provides verify and commit semantics.
- Supports configuring and synchronizing port profile configurations.
- Provides an import command to migrate existing vPC configurations to a switch profile.

Switch Profile Configuration Modes

The switch profile feature includes the following configuration modes:

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

Configuration Synchronization Mode

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

Switch Profile Mode

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

Switch Profile Import Mode

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

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

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

Configuration Validation

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

- Mutual Exclusion Checks
- Merge Checks

Mutual Exclusion Checks

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

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

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

The following exceptions apply to the mutual exclusion policy:

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



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

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

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

- Shutdown/no shutdown
- System QoS

Merge Checks

Merge checks are done on the peer switch that is receiving a configuration. The merge checks ensure that the received configuration does not conflict with the switch profile configuration that already exists on the receiving switch. The merge check occurs during the merge or commit process. Errors are reported as merge failures and must be manually corrected.

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

Software Upgrades and Downgrades with Switch Profiles

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

When you upgrade from an earlier release, you have the option to move some of the running-configuration commands to a switch profile. The **import** command allows you to import relevant switch profile commands. An upgrade can occur if there are buffered configurations (uncommitted); however, the uncommitted configurations are lost.

Prerequisites for Switch Profiles

Switch profiles have the following prerequisites:

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

Guidelines and Limitations for Switch Profiles

Consider the following guidelines and limitations when configuring switch profiles:

- You can only enable configuration synchronization using the mgmt0 interface.
- Configuration synchronization is performed using the mgmt 0 interface and cannot be performed using a management SVI.
- You must configure synchronized peers with the same switch profile name.
- Commands that are qualified for a switch profile configuration are allowed to be configured in the configuration switch profile (config-sync-sp) mode.
- One switch profile session can be in progress at a time. Attempts to start another session will fail.
- Supported command changes made from the configuration terminal mode are blocked when a switch profile session is in progress. You should not make unsupported command changes from the configuration terminal mode when a switch profile session is in progress.
- When you enter the **commit** command and a peer switch is reachable, the configuration is applied to both peer switches or neither switch. If there is a commit failure, the commands remain in the switch profile buffer. You can then make necessary corrections and try the commit again.
- Once a port channel is configured using switch profile mode, it cannot be configured using global configuration (config terminal) mode.



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

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

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

- Shutdown and no shutdown can be configured in either global configuration mode or switch profile mode.
- If a port channel is created in global configuration mode, channel groups including member interfaces must also be created using global configuration mode.
- Port channels that are configured within switch profile mode may have members both inside and outside of a switch profile.
- If you want to import a member interface to a switch profile, the port channel including the member interface must also be present within the switch profile.
- Defaulting an interface does not remove a channel group from the config-sync configuration for that interface. You must apply the **no channel-group** command on the interface or include the port channel in the config-sync configuration to prevent any conflicting configurations from being pushed by the config-sync module.

Guidelines for Synchronizing After Connectivity Loss

- Synchronizing configurations after mgmt0 interface connectivity loss—When mgmt0 interface connectivity is lost and configuration changes are required, apply the configuration changes on both switches using the switch profile. When connectivity to the mgmt0 interface is restored, both switches synchronize automatically.

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

Configuring Switch Profiles

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

Before you begin

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

SUMMARY STEPS

1. **configure terminal**

2. **cfs ipv4 distribute**
3. **config sync**
4. **switch-profile *name***
5. **sync-peers destination *IP-address***
6. (Optional) **show switch-profile *name* status**
7. **exit**
8. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	cfs ipv4 distribute Example: <pre>switch(config)# cfs ipv4 distribute switch(config)#</pre>	Enables CFS distribution between the peer switches.
Step 3	config sync Example: <pre>switch# config sync switch(config-sync)#</pre>	Enters configuration synchronization mode.
Step 4	switch-profile <i>name</i> Example: <pre>switch(config-sync)# switch-profile abc switch(config-sync-sp)#</pre>	Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode.
Step 5	sync-peers destination <i>IP-address</i> Example: <pre>switch(config-sync-sp)# sync-peers destination 10.1.1.1 switch(config-sync-sp)#</pre>	Configures the peer switch.
Step 6	(Optional) show switch-profile <i>name</i> status Example: <pre>switch(config-sync-sp)# show switch-profile abc status switch(config-sync-sp)#</pre>	Views the switch profile on the local switch and the peer switch information.
Step 7	exit Example: <pre>switch(config-sync-sp)# exit switch#</pre>	Exits the switch profile configuration mode and returns to EXEC mode.

	Command or Action	Purpose
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.

Example

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

```
switch# configuration terminal
switch(config)# cfs ipv4 distribute
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# sync-peers destination 10.1.1.1
switch(config-sync-sp)# show switch-profile abc status
Start-time: 15801 usecs after Mon Aug 23 06:21:08 2010
End-time: 6480 usecs after Mon Aug 23 06:21:13 2010

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

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

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

Adding a Switch to a Switch Profile

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

Follow these guidelines when adding switches:

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

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

Before you begin

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

SUMMARY STEPS

1. **config sync**
2. **switch-profile** *name*
3. **sync-peers destination** *destination IP*
4. **exit**
5. (Optional) **show switch-profile peer**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	config sync Example: switch# config sync switch(config-sync)#	Enters configuration synchronization mode.
Step 2	switch-profile <i>name</i> Example: switch(config-sync)# switch-profile abc switch(config-sync-sp)#	Configures switch profile, names the switch profile, and enters switch profile synchronization configuration mode.
Step 3	sync-peers destination <i>destination IP</i> Example: switch(config-sync-sp)# sync-peers destination 10.1.1.1 switch(config-sync-sp)#	Adds a switch to the switch profile.
Step 4	exit Example: switch(config-sync-sp)# exit switch#	Exits switch profile configuration mode.
Step 5	(Optional) show switch-profile peer Example: switch# show switch-profile peer	Displays the switch profile peer configuration.
Step 6	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Adding or Modifying Switch Profile Commands

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

Follow these guidelines when adding or modifying switch profile commands:

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

Before you begin

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

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

SUMMARY STEPS

1. **config sync**
2. **switch-profile name**
3. *Command argument*
4. (Optional) **show switch-profile name buffer**
5. **verify**
6. **commit**
7. (Optional) **show switch-profile name status**
8. **exit**
9. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	config sync Example:	Enters configuration synchronization mode.

	Command or Action	Purpose
	switch# config sync switch(config-sync)#	
Step 2	switch-profile name Example: switch(config-sync)# switch-profile abc switch(config-sync-sp)#	Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode.
Step 3	<i>Command argument</i> Example: switch(config-sync-sp)# interface Port-channel100 switch(config-sync-sp-if)# speed 1000 switch(config-sync-sp-if)# interface Ethernet1/1 switch(config-sync-sp-if)# speed 1000 switch(config-sync-sp-if)# channel-group 100	Adds a command to the switch profile.
Step 4	(Optional) show switch-profile name buffer Example: switch(config-sync-sp)# show switch-profile abc buffer switch(config-sync-sp)#	Displays the configuration commands in the switch profile buffer.
Step 5	verify Example: switch(config-sync-sp)# verify	Verifies the commands in the switch profile buffer.
Step 6	commit Example: switch(config-sync-sp)# commit	Saves the commands in the switch profile and synchronizes the configuration with the peer switch.
Step 7	(Optional) show switch-profile name status Example: switch(config-sync-sp)# show switch-profile abc status switch(config-sync-sp)#	Displays the status of the switch profile on the local switch and the status on the peer switch.
Step 8	exit Example: switch(config-sync-sp)# exit switch#	Exits the switch profile configuration mode.
Step 9	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

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

```
switch# configuration terminal
switch(config)# cfs ipv4 distribute
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# sync-peers destination 10.1.1.1
switch(config-sync-sp)# interface port-channel100
switch(config-sync-sp-if)# speed 1000
switch(config-sync-sp-if)# interface Ethernet1/1
switch(config-sync-sp-if)# speed 1000
switch(config-sync-sp-if)# channel-group 100
switch(config-sync-sp)# verify
switch(config-sync-sp)# commit
switch(config-sync-sp)# exit
switch#
```

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

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

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

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

Importing a Switch Profile

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

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

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

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

SUMMARY STEPS

1. **config sync**
2. **switch-profile** *name*
3. **import** {*interface port/slot* | *running-config* [**exclude interface ethernet**]}
4. **commit**
5. (Optional) **abort**
6. **exit**
7. (Optional) **show switch-profile**
8. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	config sync Example: <pre>switch# config sync switch(config-sync) #</pre>	Enters configuration synchronization mode.
Step 2	switch-profile <i>name</i> Example: <pre>switch(config-sync) # switch-profile abc switch(config-sync-sp) #</pre>	Configures the switch profile, names the switch profile, and enters switch profile synchronization configuration mode.
Step 3	import { <i>interface port/slot</i> <i>running-config</i> [exclude interface ethernet]} Example: <pre>switch(config-sync-sp) # import ethernet 1/2 switch(config-sync-sp-import) #</pre>	Identifies the commands that you want to import and enters switch profile import mode. <ul style="list-style-type: none"> • <CR>—Adds selected commands. • interface—Adds the supported commands for a specified interface. • running-config—Adds supported system-level commands. • running-config exclude interface ethernet—Adds supported system-level commands excluding the physical interface commands.
Step 4	commit Example: <pre>switch(config-sync-sp-import) # commit</pre>	Imports the commands and saves the commands to the switch profile.
Step 5	(Optional) abort Example: <pre>switch(config-sync-sp-import) # abort</pre>	Aborts the import process.

	Command or Action	Purpose
Step 6	exit Example: <pre>switch(config-sync-sp) # exit switch#</pre>	Exits switch profile import mode.
Step 7	(Optional) show switch-profile Example: <pre>switch# show switch-profile</pre>	Displays the switch profile configuration.
Step 8	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Example

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

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

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

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

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

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

Verifying Commands in a Switch Profile

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

SUMMARY STEPS

1. **config sync**
2. **switch-profile** *name*
3. **verify**
4. **exit**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

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

Isolating a Peer Switch

You can isolate a peer switch in order to make changes to a switch profile. This process can be used when you want to block a configuration synchronization or when you want to debug configurations.

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

To temporarily isolate a peer switch, follow these steps:

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

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

Deleting a Switch Profile

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

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

SUMMARY STEPS

1. **config sync**
2. **no switch-profile name {all-config | local-config}**
3. **exit**
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	config sync Example: <pre>switch# config sync switch(config-sync)#</pre>	Enters configuration synchronization mode.
Step 2	no switch-profile name {all-config local-config} Example: <pre>switch(config-sync)# no switch-profile abc local-config switch(config-sync-sp)#</pre>	Deletes the switch profile as follows: <ul style="list-style-type: none"> • all-config—Deletes the switch profile on the local and peer switch. If the peer switch is not reachable, only the local switch profile is deleted. • local-config—Deletes the switch profile and local configuration.
Step 3	exit Example: <pre>switch(config-sync-sp)# exit switch#</pre>	Exits configuration synchronization mode.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Deleting a Switch from a Switch Profile

You can delete a switch from a switch profile.

SUMMARY STEPS

1. **config sync**
2. **switch-profile** *name*
3. **no sync-peers destination** *destination IP*
4. **exit**
5. (Optional) **show switch-profile**
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

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

Displaying the Switch Profile Buffer

SUMMARY STEPS

1. switch# **configure sync**
2. switch(config-sync) # **switch-profile profile-name**
3. switch(config-sync-sp) # **show switch-profile profile-name buffer**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure sync	Enters configuration synchronization mode.
Step 2	switch(config-sync) # switch-profile profile-name	Enters switch profile synchronization configuration mode for the specified switch profile.
Step 3	switch(config-sync-sp) # show switch-profile profile-name buffer	Enters interface switch profile synchronization configuration mode for the specified interface.

Example

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

```
switch# configure sync
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# show switch-profile sp buffer
-----
Seq-no  Command
-----
1      vlan 101
1.1    ip igmp snooping querier 10.101.1.1
2      mac address-table static 0000.0000.0001 vlan 101 drop
3      interface Ethernet1/2
3.1    switchport mode trunk
3.2    switchport trunk allowed vlan 101

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

Synchronizing Configurations After a Switch Reboot

If a Cisco Nexus 3600 platform switch reboots while a new configuration is being committed on a peer switch using a switch profile, complete the following steps to synchronize the peer switches after reload:

SUMMARY STEPS

1. Reapply configurations that were changed on the peer switch during the reboot.
2. Enter the **commit** command.
3. Verify that the configuration is applied correctly and both peers are back synchronized.

DETAILED STEPS

-
- Step 1** Reapply configurations that were changed on the peer switch during the reboot.
- Step 2** Enter the **commit** command.
- Step 3** Verify that the configuration is applied correctly and both peers are back synchronized.
-

Example

Switch Profile Configuration show Commands

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

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

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

Supported Switch Profile Commands

The following switch profile commands are supported:

- **logging event link-status default**
- **[no] vlan *vlan-range***
- **ip access-list *acl-name***
- **policy-map type network-qos jumbo-frames**
 - **class type network-qos class-default**
 - **mtu *mtu value***
- **system qos**
 - **service-policy type network-qos jumbo-frames**
- **vlan configuration *vlan id***
 - **ip igmp snooping querier *ip***
- **spanning-tree port type edge default**
- **spanning-tree port type edge bpduguard default**
- **spanning-tree loopguard default**
- **no spanning-tree vlan *vlan id***
- **port-channel load-balance ethernet source-dest-port**
- **interface port-channel *number***
 - **description *text***
 - **switchport mode trunk**
 - **switchport trunk allowed vlan *vlan list***
 - **spanning-tree port type network**
 - **no negotiate auto**
 - **vpc peer-link**
- **interface port-channel *number***
 - **switchport access vlan *vlan id***
 - **spanning-tree port type edge**
 - **speed 10000**

- `vpc number`
- `interface ethernetx/y`
 - `switchport access vlan vlanid`
 - `spanning-tree port type edge`
 - `channel-group number mode active`

Configuration Examples for Switch Profiles

Creating a Switch Profile on a Local and Peer Switch Example

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

SUMMARY STEPS

1. Enable CFSOIP distribution on the local and the peer switch.
2. Create a switch profile on the local and the peer switch.
3. Verify that the switch profiles are the same on the local and the peer switch.
4. Verify the commands in the switch profile.
5. Apply the commands to the switch profile and to synchronize the configurations between the local and the peer switch.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Enable CFSOIP distribution on the local and the peer switch. Example: switch# <code>configuration terminal</code> switch(config)# <code>cfs ipv4 distribute</code>	
Step 2	Create a switch profile on the local and the peer switch. Example: switch(config-sync)# <code>switch-profile abc</code> switch(config-sync-sp)# <code>sync-peers destination 10.1.1.1</code>	
Step 3	Verify that the switch profiles are the same on the local and the peer switch. Example: switch(config-sync-sp)# <code>show switch-profile abc status</code> Start-time: 15801 usecs after Mon Aug 23 06:21:08 2010	

	Command or Action	Purpose
	<pre>End-time: 6480 usecs after Mon Aug 23 06:21:13 2010 Profile-Revision: 1 Session-type: Initial-Exchange Peer-triggered: Yes Profile-status: Sync Success Local information: ----- Status: Commit Success Error(s): Peer information: ----- IP-address: 10.1.1.1 Sync-status: In Sync. Status: Commit Success Error(s):</pre>	
Step 4	<p>Verify the commands in the switch profile.</p> <p>Example:</p> <pre>switch(config-sync-sp-if)# verify Verification Successful</pre>	
Step 5	<p>Apply the commands to the switch profile and to synchronize the configurations between the local and the peer switch.</p> <p>Example:</p> <pre>switch(config-sync-sp)# commit Commit Successful switch(config-sync)#</pre>	

Verifying the Synchronization Status Example

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

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

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

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

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

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

Displaying the Running Configuration

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

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

switch(config-sync)#
```

Displaying the Switch Profile Synchronization Between Local and Peer Switches

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

```
switch1# show switch-profile sp status

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

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

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

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

switch1#

switch2# show switch-profile sp status

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

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

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

Peer information:
-----
```

```

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

```

```
switch2#
```

Displaying Verify and Commit on Local and Peer Switches

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

```

switch1# configure sync
Enter configuration commands, one per line. End with CNTL/Z.
switch1(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch1(config-sync-sp)# interface ethernet1/1
switch1(config-sync-sp-if)# description foo
switch1(config-sync-sp-if)# verify
Verification Successful
switch1(config-sync-sp)# commit
Commit Successful
switch1(config-sync)# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.52
  interface Ethernet1/1
    description foo
switch1(config-sync)# show switch-profile sp status

```

```

Start-time: 171513 usecs after Wed Aug 11 17:51:28 2010
End-time: 676451 usecs after Wed Aug 11 17:51:43 2010

```

```

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

```

```

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

```

```

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

```

```
switch1(config-sync)#
```

```

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

```

```

Start-time: 265716 usecs after Wed Aug 11 16:51:28 2010
End-time: 734702 usecs after Wed Aug 11 16:51:43 2010

```

```

Profile-Revision: 3
Session-type: Commit

```

```
Peer-triggered: Yes
Profile-status: Sync Success
```

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

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

```
switch2#
```

Successful and Unsuccessful Synchronization Examples

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

```
switch# show switch-profile abc peer

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

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

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

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

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

```
switch# configure sync
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# vlan 101
switch(config-sync-sp-vlan)# ip igmp snooping querier 10.101.1.1
switch(config-sync-sp-vlan)# exit
switch(config-sync-sp)# mac address-table static 0000.0000.0001 vlan 101 drop
switch(config-sync-sp)# interface ethernet1/2
switch(config-sync-sp-if)# switchport mode trunk
switch(config-sync-sp-if)# switchport trunk allowed vlan 101
switch(config-sync-sp-if)# exit
switch(config-sync-sp)# show switch-profile sp buffer
-----
Seq-no  Command
-----
```

```
1      vlan 101
1.1    ip igmp snooping querier 10.101.1.1
2      mac address-table static 0000.0000.0001 vlan 101 drop
3      interface Ethernet1/2
3.1    switchport mode trunk
3.2    switchport trunk allowed vlan 101

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

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

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




CHAPTER 5

Configuring PTP

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

This chapter includes the following sections:

- [About PTP, on page 39](#)
- [PTP Device Types, on page 39](#)
- [PTP Process, on page 40](#)
- [High Availability for PTP, on page 41](#)
- [Guidelines and Limitations for PTP, on page 41](#)
- [Default Settings for PTP, on page 41](#)
- [Configuring PTP, on page 42](#)

About PTP

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

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

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

PTP Device Types

The following clocks are common PTP devices:

Ordinary clock

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

Boundary clock

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

Transparent clock

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

There are two kinds of transparent clocks:

End-to-end transparent clock

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

Peer-to-peer transparent clock

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



Note PTP operates only in boundary clock mode. We recommend that you deploy a Grand Master Clock (10 MHz) upstream. The servers contain clocks that require synchronization and are connected to the switch.

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

PTP Process

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

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

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

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

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

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

High Availability for PTP

Stateful restarts are not supported for PTP.

Guidelines and Limitations for PTP

- For Cisco Nexus 3600 Series switches, PTP clock correction is expected to be in the 3-digit range, from 100 to 999 nanoseconds.
- PTP operates only in boundary clock mode. End-to-end transparent clock and peer-to-peer transparent clock modes are not supported.
- PTP supports transport over User Datagram Protocol (UDP). Transport over Ethernet is not supported.
- PTP supports only multicast communication. Negotiated unicast communication is not supported.
- PTP is limited to a single domain per network.
- Forwarding PTP management packets is not supported.
- PTP-capable ports do not identify PTP packets and do not time-stamp or redirect those packets unless you enable PTP on those ports.
- 1 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 -2 to -5.

Default Settings for PTP

The following table lists the default settings for PTP parameters.

Table 2: Default PTP Parameters

Parameters	Default
PTP	Disabled
PTP version	2

Parameters	Default
PTP domain	0
PTP priority 1 value when advertising the clock	255
PTP priority 2 value when advertising the clock	255
PTP announce interval	1 log second
PTP sync interval	– 2 log seconds
PTP announce timeout	3 announce intervals
PTP minimum delay request interval	0 log seconds
PTP VLAN	1

Configuring PTP

Configuring PTP Globally

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config) # **[no] feature ptp**
3. switch(config) # **[no] ptp source ip-address [vrf vrf]**
4. (Optional) switch(config) # **[no] ptp domain number**
5. (Optional) switch(config) # **[no] ptp priority1 value**
6. (Optional) switch(config) # **[no] ptp priority2 value**
7. (Optional) switch(config) # **show ptp brief**
8. (Optional) switch(config) # **show ptp clock**
9. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	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 <i>ip-address</i> can be in IPv4 format.

	Command or Action	Purpose
Step 4	(Optional) switch(config) # [no] ptp domain <i>number</i>	Configures the domain number to use for this clock. PTP domains allow you to use multiple independent PTP clocking subdomains on a single network. The range for the <i>number</i> is from 0 to 128.
Step 5	(Optional) switch(config) # [no] ptp priority1 <i>value</i>	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. The range for the <i>value</i> is from 0 to 255.
Step 6	(Optional) switch(config) # [no] ptp priority2 <i>value</i>	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. The range for the <i>value</i> is from 0 to 255.
Step 7	(Optional) switch(config) # show ptp brief	Displays the PTP status.
Step 8	(Optional) switch(config) # show ptp clock	Displays the properties of the local clock.
Step 9	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

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

```

Mean Path Delay : 0
Steps removed : 0
Local clock time:Sun Jul 3 14:13:24 2011
switch(config)#

```

Configuring PTP on an Interface

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

Before you begin

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # interface ethernet slot/port	Specifies the interface on which you are enabling PTP and enters interface configuration mode.
Step 3	switch(config-if) # [no] feature ptp	Enables or disables PTP on an interface.
Step 4	(Optional) switch(config-if) # [no] ptp announce {interval log seconds timeout count}	Configures the interval between PTP announce messages on an interface or the number of PTP intervals before a timeout occurs on an interface. The range for the PTP announcement interval is from 0 to 4 seconds, and the range for the interval timeout is from 2 to 10.
Step 5	(Optional) switch(config-if) # [no] ptp delay request minimum interval log seconds	Configures the minimum interval allowed between PTP delay-request messages when the port is in the master state. The range is from log(-6) to log(1) seconds. Where, log(-2) = 2 frames per second.

	Command or Action	Purpose
Step 6	(Optional) switch(config-if) # [no] ptp sync interval <i>log seconds</i>	Configures the interval between PTP synchronization messages on an interface. The range for the PTP synchronization interval is from -6 log second to 1 second.
Step 7	(Optional) switch(config-if) # [no] ptp vlan <i>vlan-id</i>	Specifies the VLAN for the interface where PTP is being enabled. You can only enable PTP on one VLAN on an interface. The range is from 1 to 4094.
Step 8	(Optional) switch(config-if) # show ptp brief	Displays the PTP status.
Step 9	(Optional) switch(config-if) # show ptp port interface <i>interface slot/port</i>	Displays the status of the PTP port.
Step 10	(Optional) switch(config-if)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

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

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 <i>slot/port</i>	Displays the status of the PTP port on the switch.



CHAPTER 6

Configuring NTP

This chapter contains the following sections:

- [Information About NTP, on page 47](#)
- [NTP as Time Server, on page 48](#)
- [Distributing NTP Using CFS, on page 48](#)
- [Clock Manager, on page 48](#)
- [High Availability, on page 48](#)
- [Virtualization Support, on page 48](#)
- [Prerequisites for NTP, on page 49](#)
- [Guidelines and Limitations for NTP, on page 49](#)
- [Default Settings, on page 50](#)
- [Configuring NTP, on page 50](#)
- [Verifying the NTP Configuration, on page 63](#)
- [Configuration Examples for NTP, on page 64](#)

Information About NTP

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

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

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

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

Before synchronizing, NTP compares the time reported by several network devices and does not synchronize with one that is significantly different, even if it is a stratum 1. Because Cisco NX-OS cannot connect to a radio or atomic clock and act as a stratum 1 server, we recommend that you use the public NTP servers

available on the Internet. If the network is isolated from the Internet, Cisco NX-OS allows you to configure the time as though it were synchronized through NTP, even though it was not.



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

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

NTP as Time Server

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

Distributing NTP Using CFS

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

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

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

Clock Manager

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

Multiple time synchronization protocol, such as NTP might be running in the system.

High Availability

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

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

Virtualization Support

NTP recognizes virtual routing and forwarding (VRF) instances. NTP uses the default VRF if you do not configure a specific VRF for the NTP server and NTP peer.

Prerequisites for NTP

NTP has the following prerequisites:

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

Guidelines and Limitations for NTP

NTP has the following configuration guidelines and limitations:

- The **show ntp session status** CLI command does not show the last action timestamp, the last action, the last action result, and the last action failure reason.
- NTP server functionality is supported.
- You should have a peer association with another device only when you are sure that your clock is reliable (which means that you are a client of a reliable NTP server).
- A peer that is configured alone takes on the role of a server and should be used as a backup. If you have two servers, you can configure several devices to point to one server and the remaining devices to point to the other server. You can then configure a peer association between these two servers to create a more reliable NTP configuration.
- If you have only one server, you should configure all the devices as clients to that server.
- You can configure up to 64 NTP entities (servers and peers).
- If CFS is disabled for NTP, NTP does not distribute any configuration and does not accept a distribution from other devices in the network.
- After CFS distribution is enabled for NTP, the entry of an NTP configuration command locks the network for NTP configuration until a **commit** command is entered. During the lock, no changes can be made to the NTP configuration by any other device in the network except the device that initiated the lock.
- If you use CFS to distribute NTP, all devices in the network should have the same VRFs configured as you use for NTP.
- If you configure NTP in a VRF, ensure that the NTP server and peers can reach each other through the configured VRFs.
- You must manually distribute NTP authentication keys on the NTP server and Cisco NX-OS devices across the network.
- Use NTP broadcast or multicast associations when time accuracy and reliability requirements are modest, your network is localized, and the network has more than 20 clients. We recommend that you use NTP broadcast or multicast associations in networks that have limited bandwidth, system memory, or CPU resources.
- A maximum of four ACLs can be configured for a single NTP access group.



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

Default Settings

The following are the default settings for NTP parameters.

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

Configuring NTP

Enabling or Disabling NTP on an Interface

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface** *type slot/port*
3. switch(config-if)# **[no] ntp disable** {ip | ipv6}
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# interface <i>type slot/port</i>	Enters interface configuration mode.

	Command or Action	Purpose
Step 3	switch(config-if)# [no] ntp disable {ip ipv6}	Disables NTP IPv4 or IPv6 on the specified interface. Use the no form of this command to reenable NTP on the interface.
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 or disable NTP on an interface:

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

Configuring the Device as an Authoritative NTP Server

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] ntp master [stratum]	Configures the device as an authoritative NTP server. You can specify a different stratum level from which NTP clients get their time synchronized. The range is from 1 to 15.
Step 3	(Optional) show running-config ntp	Displays the NTP configuration.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

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

Configuring an NTP Server and Peer

You can configure an NTP server and peer.

Before you begin

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

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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# [no] ntp server { <i>ip-address</i> <i>ipv6-address</i> <i>dns-name</i> } [key <i>key-id</i>] [maxpoll <i>max-poll</i>] [minpoll <i>min-poll</i>] [prefer] [use-vrf <i>vrf-name</i>]	<p>Forms an association with a server.</p> <p>Use the key keyword to configure a key to be used while communicating with the NTP server.</p> <p>The range for the <i>key-id</i> argument is from 1 to 65535.</p> <p>Use the maxpoll and minpoll keywords to configure the maximum and minimum intervals in which to poll a server. The range for the <i>max-poll</i> and <i>min-poll</i> arguments is from 4 to 16 (configured as powers of 2, so effectively 16 to 65536 seconds), and the default values are 6 and 4, respectively (<i>maxpoll</i> default = 64 seconds, <i>minpoll</i> default = 16 seconds).</p> <p>Use the prefer keyword to make this the preferred NTP server for the device.</p> <p>Use the use-vrf keyword to configure the NTP server to communicate over the specified VRF.</p>

	Command or Action	Purpose
		<p>The <i>vrf-name</i> argument can be default, management, or any case-sensitive alphanumeric string up to 32 characters.</p> <p>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.</p>
Step 3	switch(config)# [no] ntp peer { <i>ip-address</i> <i>ipv6-address</i> <i>dns-name</i> } [key <i>key-id</i>] [maxpoll <i>max-poll</i>] [minpoll <i>min-poll</i>] [prefer] [use-vrf <i>vrf-name</i>]	<p>Forms an association with a peer. You can specify multiple peer associations.</p> <p>Use the key keyword to configure a key to be used while communicating with the NTP peer. The range for the <i>key-id</i> argument is from 1 to 65535.</p> <p>Use the maxpoll and minpoll keywords to configure the maximum and minimum intervals in which to poll a server. The range for the <i>max-poll</i> and <i>min-poll</i> arguments is from 4 to 17 (configured as powers of 2, so effectively 16 to 131072 seconds), and the default values are 6 and 4, respectively (<i>maxpoll</i> default = 64 seconds, <i>minpoll</i> default = 16 seconds).</p> <p>Use the prefer keyword to make this the preferred NTP peer for the device.</p> <p>Use the use-vrf keyword to configure the NTP peer to communicate over the specified VRF. The <i>vrf-name</i> argument can be default, management, or any case-sensitive alphanumeric string up to 32 characters.</p>
Step 4	(Optional) switch(config)# show ntp peers	<p>Displays the configured server and peers.</p> <p>Note A domain name is resolved only when you have a DNS server configured.</p>
Step 5	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Configuring NTP Authentication

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

Before you begin

Authentication for NTP servers and NTP peers is configured on a per-association basis using the **key** keyword on each **ntp server** and **ntp peer** command. Make sure that you configured all NTP server and peer associations

with the authentication keys that you plan to specify in this procedure. Any **ntp server** or **ntp peer** commands that do not specify the **key** keyword will continue to operate without authentication.

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

	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 (defined in Step 2) that an unconfigured remote symmetric, broadcast, and multicast time source must provide in its NTP packets in order for the device to synchronize to it. The range for trusted keys is from 1 to 65535. This command provides protection against accidentally synchronizing the device to a time source that is not trusted. This command does not affect time sources configured with the ntp server and ntp peer configuration comments.
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# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp authentication-key 42 md5 aNiceKey
switch(config)# ntp server 10.1.1.1 key 42
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 match-all | {peer | serve | serve-only | query-only} access-list-name**
3. switch(config)# **show ntp access-groups**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# [no] ntp access-group match-all {peer serve serve-only query-only} access-list-name	<p>Creates or removes an access group to control NTP access and applies a basic IP access list.</p> <p>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.</p> <ul style="list-style-type: none"> • 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.

	Command or Action	Purpose
		<ul style="list-style-type: none"> The serve-only keyword enables the device to receive only time requests from servers specified in the access list. The query-only keyword enables the device to receive only NTP control queries from the servers specified in the access list. The match-all keyword enables the access group options to be scanned in the following order, from least restrictive to most restrictive: peer, serve, serve-only, query-only. If the incoming packet does not match the ACL in the peer access group, it goes to the serve access group to be processed. If the packet does not match the ACL in the serve access group, it goes to the serve-only access group, and so on.
Step 3	switch(config)# show ntp access-groups	(Optional) Displays the NTP access group configuration.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

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

Configuring the NTP Source IP Address

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

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **ntp source ip-address**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] ntp source <i>ip-address</i>	Configures the source IP address for all NTP packets. The <i>ip-address</i> can be in IPv4 or IPv6 format.

Example

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

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

Configuring the NTP Source Interface

You can configure NTP to use a specific interface.

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **ntp source-interface** *interface*

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] ntp source-interface <i>interface</i>	Configures the source interface for all NTP packets. The following list contains the valid values for <i>interface</i> . <ul style="list-style-type: none"> • ethernet • loopback • mgmt • port-channel • vlan

Example

This example shows how to configure the NTP source interface:

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

Configuring an NTP Broadcast Server

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface** *type slot/port*
3. switch(config-if)# **[no] ntp broadcast** [*destination ip-address*] [**key** *key-id*] [*version number*]
4. switch(config-if)# **exit**
5. (Optional) switch(config)# **[no] ntp broadcastdelay** *delay*
6. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# interface <i>type slot/port</i>	Enters interface configuration mode.
Step 3	switch(config-if)# [no] ntp broadcast [<i>destination ip-address</i>] [key <i>key-id</i>] [<i>version number</i>]	Enables an NTP IPv4 broadcast server on the specified interface. <ul style="list-style-type: none"> • destination <i>ip-address</i>—Configures the broadcast destination IP address. • key <i>key-id</i>—Configures the broadcast authentication key number. The range is from 1 to 65535. • <i>version number</i>—Configures the NTP version. The range is from 2 to 4.
Step 4	switch(config-if)# exit	Exits interface configuration mode.
Step 5	(Optional) switch(config)# [no] ntp broadcastdelay <i>delay</i>	Configures the estimated broadcast round-trip delay in microseconds. The range is from 1 to 999999.
Step 6	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure an NTP broadcast server:

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

Configuring an NTP Multicast Server

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface** *type slot/port*
3. switch(config-if)# **[no] ntp multicast** [*ipv4-address* | *ipv6-address*] [**key** *key-id*] [*ttl value*] [*version number*]
4. (Optional) switch(config-if)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# interface <i>type slot/port</i>	Enters interface configuration mode.
Step 3	switch(config-if)# [no] ntp multicast [<i>ipv4-address</i> <i>ipv6-address</i>] [key <i>key-id</i>] [<i>ttl value</i>] [<i>version number</i>]	Enables an NTP IPv4 or IPv6 multicast server on the specified interface. <ul style="list-style-type: none"> • <i>ipv4-address</i> or <i>ipv6-address</i>— Multicast IPv4 or IPv6 address. • key <i>key-id</i>—Configures the broadcast authentication key number. The range is from 1 to 65535. • <i>ttl value</i>—Time-to-live value of the multicast packets. The range is from 1 to 255. • <i>version number</i>—NTP version. The range is from 2 to 4.
Step 4	(Optional) switch(config-if)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

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

Configuring an NTP Multicast Client

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

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface** *type slot/port*
3. switch(config-if)# **[no] ntp multicast client** [*ipv4-address* | *ipv6-address*]
4. (Optional) switch(config-if)# **copy running-config startup-config**

DETAILED STEPS

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

Example

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

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

Configuring NTP Logging

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

Before you begin

Make sure that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

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

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

Example

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

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

Enabling CFS Distribution for NTP

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

Before you begin

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

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

	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.

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

Example

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

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

Committing NTP Configuration Changes

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

SUMMARY STEPS

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

DETAILED STEPS

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

Discarding NTP Configuration Changes

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

SUMMARY STEPS

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

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	switch(config)# ntp abort	Discards the NTP configuration changes in the pending database and releases the CFS lock. Use this command on the device where you started the NTP configuration.

Releasing the CFS Session Lock

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

SUMMARY STEPS

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

DETAILED STEPS

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

Verifying the NTP Configuration

Command	Purpose
show ntp access-groups	Displays the NTP access group configuration.
show ntp authentication-keys	Displays the configured NTP authentication keys.
show ntp authentication-status	Displays the status of NTP authentication.
show ntp logging-status	Displays the NTP logging status.
show ntp peer-status	Displays the status for all NTP servers and peers.
show ntp peer	Displays all the NTP peers.
show ntp pending	Displays the temporary CFS database for NTP.
show ntp pending-diff	Displays the difference between the pending CFS database and the current NTP configuration.
show ntp rts-update	Displays the RTS update status.
show ntp session status	Displays the NTP CFS distribution session information.

Command	Purpose
<code>show ntp source</code>	Displays the configured NTP source IP address.
<code>show ntp source-interface</code>	Displays the configured NTP source interface.
<code>show ntp statistics {io local memory peer {ipaddr {ipv4-addr} name peer-name}}</code>	Displays the NTP statistics.
<code>show ntp status</code>	Displays the NTP CFS distribution status.
<code>show ntp trusted-keys</code>	Displays the configured NTP trusted keys.
<code>show running-config ntp</code>	Displays NTP information.

Configuration Examples for NTP

Configuration Examples for NTP

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

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp server 192.0.2.105 key 42
switch(config)# ntp peer 192.0.2.105
switch(config)# show ntp peers
-----
Peer IP Address Serv/Peer
-----
192.0.2.100 Peer (configured)
192.0.2.105 Server (configured)
switch(config)# ntp authentication-key 42 md5 aNiceKey
switch(config)# show ntp authentication-keys
-----
Auth key MD5 String
-----
42 aNicekey
switch(config)# ntp trusted-key 42
switch(config)# show ntp trusted-keys
Trusted Keys:
42
switch(config)# ntp authenticate
switch(config)# show ntp authentication-status
Authentication enabled.
switch(config)# ntp logging
switch(config)# show ntp logging
NTP logging enabled.
switch(config)# copy running-config startup-config
[#####] 100%
switch(config)#
```

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

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

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

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




CHAPTER 7

Configuring Session Manager

This chapter contains the following sections:

- [About Session Manager, on page 67](#)
- [Guidelines and Limitations for Session Manager, on page 67](#)
- [Configuring Session Manager, on page 68](#)
- [Verifying the Session Manager Configuration, on page 70](#)

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

	Command or Action	Purpose
Step 1	switch# configure session <i>name</i>	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 [<i>name</i>]	Displays the contents of the session.
Step 3	(Optional) switch(config-s)# save <i>location</i>	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

	Command or Action	Purpose
Step 1	switch# configure session <i>name</i>	Creates a configuration session and enters session configuration mode. The name can be any alphanumeric string.

	Command or Action	Purpose
Step 2	switch(config-s)# ip access-list <i>name</i>	Creates an ACL.
Step 3	(Optional) switch(config-s-acl)# permit <i>protocol source destination</i>	Adds a permit statement to the ACL.
Step 4	switch(config-s-acl)# interface <i>interface-type number</i>	Enters interface configuration mode.
Step 5	switch(config-s-if)# ip port access-group <i>name in</i>	Adds a port access group to the interface.
Step 6	(Optional) switch# show configuration session [<i>name</i>]	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 <i>location</i>	(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 [<i>name</i>]	Displays the contents of the configuration session.
show configuration session status [<i>name</i>]	Displays the status of the configuration session.
show configuration session summary	Displays a summary of all the configuration sessions.



CHAPTER 8

Configuring Smart Call Home

This chapter contains the following sections:

- [About Smart Call Home, on page 71](#)
- [Guidelines and Limitations for Smart Call Home, on page 79](#)
- [Prerequisites for Smart Call Home, on page 79](#)
- [Default Call Home Settings, on page 80](#)
- [Configuring Smart Call Home, on page 80](#)
- [Verifying the Smart Call Home Configuration, on page 92](#)
- [Sample Syslog Alert Notification in Full-Text Format, on page 92](#)
- [Sample Syslog Alert Notification in XML Format, on page 93](#)

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 4: 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 5: 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 6: 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 7: 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: <i>YYYY-MM-DD HH:MM:SS</i> <i>GMT+HH:MM</i>	/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	<p>Unique device identifier (UDI) for the end device that generated the message. This field should be empty if the message is nonspecific to a device. The format is <i>type@Sid@serial</i>:</p> <ul style="list-style-type: none"> • <i>type</i> is the product model number from backplane IDPROM. • <i>@</i> is a separator character. • <i>Sid</i> is C, identifying the serial ID as a chassis serial number. • <i>serial</i> is the number identified by the Sid field. <p>An example is WS-C6509@C@12345678</p>	/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	<p>If the message is generated from the device, this is the unique device identifier (UDI) of the device.</p> <p>The format is <i>type@Sid@serial</i>:</p> <ul style="list-style-type: none"> • <i>type</i> is the product model number from backplane IDPROM. • <i>@</i> is a separator character. • <i>Sid</i> is C, identifying the serial ID as a chassis serial number. • <i>serial</i> is the number identified by the Sid field. <p>An example is WS-C6509@C@12345678</p>	/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 8: 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 9: 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 10: 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.
- Smart Call Home operates with any SMTP server.
- You can configure up to five SMTP servers for Smart Call Home.
- Link up/down syslog messages do not trigger Smart Call Home messages or alert notifications.
- Beginning with Cisco NX-OS Release 7.0(3)F3(4), the output of the **show environment fan** and **show environment power** commands indicates if the power supply fan fails. In previous releases, only the **show environment fan** command shows the failure.



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

Prerequisites for Smart Call Home

- You must have e-mail server connectivity.
- You must have access to contact name (SNMP server contact), phone, and street address information.
- You must have IP connectivity between the switch and the e-mail server.

- You must have an active service contract for the device that you are configuring.

Default Call Home Settings

Table 11: Default Call Home Parameters

Parameters	Default
Destination message size for a message sent in full text format	4000000
Destination message size for a message sent in XML format	4000000
Destination message size for a message sent in short text format	4000
SMTP server port number if no port is specified	25
Alert group association with profile	All for full-text-destination and short-text-destination profiles. The cisco-tac alert group for the CiscoTAC-1 destination profile.
Format type	XML
Call Home message level	0 (zero)

Configuring Smart Call Home

Registering for Smart Call Home

Before you begin

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

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

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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# snmp-server contact <i>sys-contact</i>	Configures the SNMP sysContact.
Step 3	switch(config)# callhome	Enters Smart Call Home configuration mode.
Step 4	switch(config-callhome)# email-contact <i>email-address</i>	Configures the e-mail address for the primary person responsible for the switch. The <i>email-address</i> 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 <i>international-phone-number</i>	Configures the phone number in international phone number format for the primary person responsible for the device. The <i>international-phone-number</i> can be up to 17

	Command or Action	Purpose
		alphanumeric characters and must be in international phone number format. Note The phone number cannot contain spaces. Use the plus (+) prefix before the number.
Step 6	switch(config-callhome)# streetaddress <i>address</i>	Configures the street address for the primary person responsible for the switch. The <i>address</i> can be up to 255 alphanumeric characters. Spaces are accepted.
Step 7	(Optional) switch(config-callhome)# contract-id <i>contract-number</i>	Configures the contract number for this switch from the service agreement. The <i>contract-number</i> can be up to 255 alphanumeric characters.
Step 8	(Optional) switch(config-callhome)# customer-id <i>customer-number</i>	Configures the customer number for this switch from the service agreement. The <i>customer-number</i> can be up to 255 alphanumeric characters.
Step 9	(Optional) switch(config-callhome)# site-id <i>site-number</i>	Configures the site number for this switch. The <i>site-number</i> can be up to 255 alphanumeric characters in free format.
Step 10	(Optional) switch(config-callhome)# switch-priority <i>number</i>	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.
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

	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 <i>group</i> email-addr <i>address</i> http <i>URL</i> transport-method { email http }} <i>profilename</i> { alert-group <i>group</i> email-addr <i>address</i> format { XML full-txt short-txt } http <i>URL</i> message-level <i>level</i> message-size <i>size</i> transport-method { email http }} full-txt-destination { alert-group <i>group</i> email-addr <i>address</i> http <i>URL</i> message-level <i>level</i> message-size <i>size</i> transport-method { email http }} short-txt-destination { alert-group <i>group</i> email-addr <i>address</i> http <i>URL</i> message-level <i>level</i> message-size <i>size</i> 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 [<i>profile name</i>]	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

	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 { <i>name</i> full-txt-destination short-txt-destination } email-addr <i>address</i>	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 { <i>name</i> full-txt-destination short-txt-destination } message-level <i>number</i>	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 <i>number</i> is from 0 to 9, where 9 is the highest severity level.

	Command or Action	Purpose
Step 5	switch(config-callhome)# destination-profile { <i>name</i> full-txt-destination short-txt-destination } message-size <i>number</i>	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 <i>name</i>]	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

	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 <i>name</i> 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

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

	Command or Action	Purpose
Step 3	switch(config-callhome)# alert-group { Configuration Diagnostic Environmental Inventory License Linecard-Hardware Supervisor-Hardware Syslog-group-port System Test } user-def-cmd <i>show-cmd</i>	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

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

	Command or Action	Purpose
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 <i>number</i> range is from 1 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

	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 [<i>interval days</i>] [<i>timeofday time</i>]	Configures periodic inventory messages. The <i>interval days</i> range is from 1 to 30 days. The default is 7 days. The <i>timeofday time</i> 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

	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

	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

	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 <i>name</i>	Displays one or more Smart Call Home destination profiles.
show callhome pending-diff	Displays the differences between the pending and running Smart Call Home configuration.
show callhome status	Displays the Smart Call Home status.
show callhome transport-email	Displays the e-mail configuration for Smart Call Home.
show callhome user-def-cmds	Displays CLI commands added to any alert groups.
show running-config [callhome callhome-all]	Displays the running configuration for Smart Call Home.
show startup-config callhome	Displays the startup configuration for Smart Call Home.
show tech-support callhome	Displays the technical support output for Smart Call Home.

Sample Syslog Alert Notification in Full-Text Format

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

```
source:MDS9000
Switch Priority:7
Device Id:WS-C6509@C@FG@07120011
Customer Id:Example.com
Contract Id:123
Site Id:San Jose
Server Id:WS-C6509@C@FG@07120011
Time of Event:2004-10-08T11:10:44
Message Name:SYSLOG_ALERT
Message Type:Syslog
Severity Level:2
System Name:10.76.100.177
Contact Name:User Name
Contact Email:person@example.com
```

```

Contact Phone:+1-408-555-1212
Street Address:#1234 Any Street, Any City, Any State, 12345
Event Description:2006 Oct 8 11:10:44 10.76.100.177 %PORT-5-IF_TRUNK_UP:
%$VLAN 1%$ Interface e2/5, vlan 1 is up
syslog_facility:PORT
start chassis information:
Affected Chassis:WS-C6509
Affected Chassis Serial Number:FG07120011
Affected Chassis Hardware Version:0.104
Affected Chassis Software Version:3.1(1)
Affected Chassis Part No:73-8607-01
end chassis information:

```

Sample Syslog Alert Notification in XML Format

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

```

From: example
Sent: Wednesday, April 25, 2007 7:20 AM
To: User (user)
Subject: System Notification From Router - syslog - 2007-04-25 14:19:55
GMT+00:00
<?xml version="1.0" encoding="UTF-8"?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.example.com/2004/01/aml-session"
soap-env:mustUnderstand="true" soap-env:role=
"http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.example.com/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.example.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.example.com/appliance/uri</aml-session:From>
<aml-session:MessageId>M2:69000101:C9D9E20B</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.example.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.example.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2007-04-25 14:19:55 GMT+00:00</aml-block:CreationDate>
<aml-block:Builder>
<aml-block:Name>Cat6500</aml-block:Name>
<aml-block:Version>2.0</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>G3:69000101:C9F9E20C</aml-block:GroupId>
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>2</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:Call Home xmlns:ch="http://www.example.com/2005/05/callhome" version="1.0">
<ch:EventTime>2007-04-25 14:19:55 GMT+00:00</ch:EventTime>
<ch:MessageDescription>03:29:29: %CLEAR-5-COUNTERS: Clear counter on all
interfaces by console</ch:MessageDescription>
<ch:Event>
<ch:Type>syslog</ch:Type>
<ch:SubType>

```

```

</ch:SubType>
<ch:Brand>Cisco Systems</ch:Brand>
<ch:Series>Catalyst 6500 Series Switches</ch:Series>
</ch:Event>
<ch:CustomerData>
<ch:UserData>
<ch:Email>person@example.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:CustomerId>12345</ch:CustomerId>
<ch:SiteId>building 1</ch:SiteId>
<ch:ContractId>abcdefg12345</ch:ContractId>
<ch:DeviceId>WS-C6509@C@69000101</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch>Name>Router</ch>Name>
<ch>Contact>
</ch>Contact>
<ch>ContactEmail>user@example.com</ch>ContactEmail>
<ch>ContactPhoneNumber>+1-408-555-1212</ch>ContactPhoneNumber>
<ch:StreetAddress>#1234 Any Street, Any City, Any State, 12345
</ch:StreetAddress>
</ch:SystemInfo>
</ch:CustomerData>
<ch:Device>
<rme:Chassis xmlns:rme="http://www.example.com/rme/4.0">
<rme:Model>WS-C6509</rme:Model>
<rme:HardwareVersion>1.0</rme:HardwareVersion>
<rme:SerialNumber>69000101</rme:SerialNumber>
<rme:AdditionalInformation>
<rme:AD name="PartNumber" value="73-3438-03 01" />
<rme:AD name="SoftwareVersion" value="4.0(20080421:012711)" />
</rme:AdditionalInformation>
</rme:Chassis>
</ch:Device>
</ch:Call Home>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging</aml-block:Name>
<aml-block:Data encoding="plain">
<![CDATA[Syslog logging: enabled (0 messages dropped, 0 messages
rate-limited, 0 flushes, 0 overruns, xml disabled, filtering disabled)
  Console logging: level debugging, 53 messages logged, xml disabled,
filtering disabled  Monitor logging: level debugging, 0 messages logged,
xml disabled,filtering disabled  Buffer logging: level debugging,
53 messages logged, xml disabled,  filtering disabled  Exception
Logging: size (4096 bytes)  Count and timestamp logging messages: disabled
  Trap logging: level informational, 72 message lines logged
Log Buffer (8192 bytes):
00:00:54: curr is 0x20000
00:00:54: RP: Currently running ROMMON from F2 region
00:01:05: %SYS-5-CONFIG_I: Configured from memory by console
00:01:09: %SYS-5-RESTART: System restarted --Cisco IOS Software,
s72033_rp Software (s72033_rp-ADVENTERPRISEK9_DBG-VM), Experimental
Version 12.2(20070421:012711) Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-07 15:54 by xxx
Firmware compiled 11-Apr-07 03:34 by integ Build [100]00:01:01: %PFREDUN-6-ACTIVE:
Initializing as ACTIVE processor for this switch00:01:01: %SYS-3-LOGGER_FLUSHED:
System was paused for 00:00:00 to ensure console debugging output.00:03:00: SP: SP:
Currently running ROMMON from F1 region00:03:07: %C6K_PLATFORM-SP-4-CONFREG_BREAK
_ENABLED: The default factory setting for config register is 0x2102.It is advisable
to retain 1 in 0x2102 as it prevents returning to ROMMON when break is issued.00:03:18:
%SYS-SP-5-RESTART: System restarted --Cisco IOS Software, s72033_sp Software

```



```

(s72033_sp-ADVENTERPRISEK9_DBG-VM), Experimental Version 12.2(20070421:012711)Copyright
(c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-07 18:00 by xxx
00:03:18: %SYS-SP-6-BOOTTIME: Time taken to reboot after reload = 339 seconds
00:03:18: %OIR-SP-6-INSFPS: Power supply inserted in slot 1
00:03:18: %C6KPWR-SP-4-PSOK: power supply 1 turned on.
00:03:18: %OIR-SP-6-INSFPS: 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, c61c2 Software (c61c2-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 17:20 by xxx
00:00:23: DFC4: Currently running ROMMON from F2 region
00:00:25: %SYS-DFC2-5-RESTART: System restarted --
Cisco IOS Software, c6slc Software (c6slc-SPDBG-VM), Experimental Version 12.2
(20070421:012711)Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 16:40 by username1
00:00:26: DFC2: Currently running ROMMON from F2 region
00:04:56: %DIAG-SP-6-RUN_MINIMUM: Module 4: Running Minimal Diagnostics...
00:00:09: DaughterBoard (Distributed Forwarding Card 3)
Firmware compiled 11-Apr-08 03:34 by integ Build [100]
slot_id is 8
00:00:31: %FLASHFS_HES-DFC8-3-BADCARD: /bootflash:: The flash card seems to
be corrupted
00:00:31: %SYS-DFC8-5-RESTART: System restarted --
Cisco DCOS Software, c61c2 Software (c61c2-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 17:20 by username1
00:00:31: DFC8: Currently running ROMMON from S (Gold) region
00:04:59: %DIAG-SP-6-RUN_MINIMUM: Module 2: Running Minimal Diagnostics...
00:05:12: %DIAG-SP-6-RUN_MINIMUM: Module 8: Running Minimal Diagnostics...
00:05:13: %DIAG-SP-6-RUN_MINIMUM: Module 1: Running Minimal Diagnostics...
00:00:24: %SYS-DFC1-5-RESTART: System restarted --
Cisco DCOS Software, c6slc Software (c6slc-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.

```

```
Compiled Thu 26-Apr-08 16:40 by username1
00:00:25: DFC1: Currently running ROMMON from F2 region
00:05:30: %DIAG-SP-6-DIAG_OK: Module 4: Passed Online Diagnostics
00:05:31: %SPAN-SP-6-SPAN_EGRESS_REPLICATION_MODE_CHANGE: Span Egress HW
Replication Mode Change Detected. Current replication mode for unused asic
session 0 is Centralized
00:05:31: %SPAN-SP-6-SPAN_EGRESS_REPLICATION_MODE_CHANGE: Span Egress HW
Replication Mode Change Detected. Current replication mode for unused asic
session 1 is Centralized
00:05:31: %OIR-SP-6-INSCARD: Card inserted in slot 4, interfaces are now online
00:06:02: %DIAG-SP-6-DIAG_OK: Module 1: Passed Online Diagnostics
00:06:03: %OIR-SP-6-INSCARD: Card inserted in slot 1, interfaces are now online
00:06:31: %DIAG-SP-6-DIAG_OK: Module 2: Passed Online Diagnostics
00:06:33: %OIR-SP-6-INSCARD: Card inserted in slot 2, interfaces are now online
00:04:30: %XDR-6-XDRIPCNOTIFY: Message not sent to slot 4/0 (4) because of IPC
error timeout. Disabling linecard. (Expected during linecard OIR)
00:06:59: %DIAG-SP-6-DIAG_OK: Module 8: Passed Online Diagnostics
00:06:59: %OIR-SP-6-DOWNGRADE_EARL: Module 8 DFC installed is not identical to
system PFC and will perform at current system operating mode.
00:07:06: %OIR-SP-6-INSCARD: Card inserted in slot 8, interfaces are now online
Router#]]>
</aml-block:Data>
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>
```



CHAPTER 9

Configuring the Scheduler

This chapter contains the following sections:

- [Information About the Scheduler, on page 97](#)
- [Guidelines and Limitations for the Scheduler, on page 98](#)
- [Default Settings for the Scheduler, on page 98](#)
- [Configuring the Scheduler, on page 99](#)
- [Verifying the Scheduler Configuration, on page 106](#)
- [Configuration Examples for the Scheduler, on page 107](#)
- [Standards for the Scheduler, on page 108](#)

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.
 - Feature id= nxos-7k-only. 3k is not a modular chassis.
 - If you have removed a module from a slot and a job for that slot 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**, and other similar commands) are specified because the job is started and conducted noninteractively.

Default Settings for the Scheduler

Table 12: Default Command Scheduler Parameters

Parameters	Default
Scheduler state	Disabled
Log file size	16 KB

Configuring the Scheduler

Enabling the Scheduler

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # feature scheduler	Enables the scheduler in the current VDC.
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

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

SUMMARY STEPS

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

DETAILED STEPS

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

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # scheduler aaa-authentication password [0 7] <i>password</i>	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 <i>name password</i> [0 7] <i>password</i>	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.

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # scheduler job name <i>name</i>	Creates a job with the specified name and enters the job configuration mode. The <i>name</i> is restricted to 31 characters.
Step 3	switch(config-job) # <i>command1</i> ; [<i>command2</i> ; <i>command3</i> ; ...	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 [<i>name</i>]	Displays the job information. The <i>name</i> 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

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

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

	Command or Action	Purpose
Step 1	<code>switch# configure terminal</code>	Enters global configuration mode.
Step 2	<code>switch(config) # no scheduler job name name</code>	Deletes the specified job and all commands defined within it. The <i>name</i> is restricted to 31 characters.
Step 3	(Optional) <code>switch(config-job) # show scheduler job [name]</code>	Displays the job information.
Step 4	(Optional) <code>switch(config-job) # copy running-config startup-config</code>	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.

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

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

	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 <i>name</i> 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 <i>name</i> 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 [[<i>day-of-week:</i>] <i>HH:</i>] <i>MM</i>	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 [[<i>day-of-month:</i>] <i>HH:</i>] <i>MM</i>	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 <i>repeat-interval</i> <i>delta-time</i> [repeat <i>repeat-interval</i>]}	Indicates the job starts periodically. The start-time format is [[[yyyy:]mmm:]dd:]HH]:MM. <ul style="list-style-type: none"> • <i>delta-time</i>— Specifies the amount of time to wait after the schedule is configured before starting a job.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • now— Specifies that the job starts two minutes from now. • repeat <i>repeat-interval</i>— Specifies the frequency at which the job is repeated.
Step 8	(Optional) switch(config-schedule) # show scheduler config	Displays the scheduler information.
Step 9	(Optional) switch(config-schedule) # copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

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

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

Clearing the Scheduler Log File

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

SUMMARY STEPS

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

DETAILED STEPS

	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

Before you begin

Confirm that you are in the correct VDC. To change the VDC, use the **switchto vdc** command.

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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # no feature scheduler	Disables the scheduler in the current VDC.
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 13: 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 (creates the filename using the current timestamp and switch name):

```
switch# configure terminal
switch(config)# scheduler job name backup-cfg
switch(config-job)# copy running-config
tftp://1.2.3.4/$(SWITCHNAME)-cfg.$(TIMESTAMP) vrf management
switch(config-job)# end
switch(config)#
```

Scheduling a Scheduler Job

This example shows how to schedule a scheduler job called backup-cfg to run daily at 1 a.m.:

```
switch# configure terminal
switch(config)# scheduler schedule name daily
switch(config-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:

- [About SNMP, on page 109](#)
- [Guidelines and Limitations for SNMP, on page 113](#)
- [Default SNMP Settings, on page 114](#)
- [Configuring SNMP, on page 114](#)
- [Configuring the SNMP Local Engine ID, on page 127](#)
- [Disabling SNMP, on page 128](#)
- [Verifying the SNMP Configuration, on page 129](#)

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 Nexus device 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 14: SNMP Security Models and Levels

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

User-Based Security Model

SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages have not been altered or destroyed in an unauthorized manner and that data sequences have not been altered to an extent greater than can occur nonmaliciously.
- Message origin authentication—Confirms that the claimed identity of the user who received the data was originated.
- Message confidentiality—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMPv3 authorizes management operations only by configured users and encrypts SNMP messages.

Cisco NX-OS uses two authentication protocols for SNMPv3:

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

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

The **priv** option offers a choice of DES or 128-bit AES encryption for SNMP security encryption. The **priv** option and the **aes-128** token indicates that this privacy password is for generating a 128-bit AES key. The AES **priv** password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 characters. If you use the localized key, you can specify a maximum of 130 characters.



Note For an SNMPv3 operation using the external AAA server, you must use AES for the privacy protocol in user configuration on the external AAA server.

CLI and SNMP User Synchronization

SNMPv3 user management can be centralized at the Access Authentication and Accounting (AAA) server level. This centralized user management allows the SNMP agent in Cisco NX-OS to leverage the user authentication service of the AAA server. Once user authentication is verified, the SNMP PDUs are processed further. Additionally, the AAA server is also used to store user group names. SNMP uses the group names to apply the access/role policy that is locally available in the switch.

Any configuration changes made to the user group, role, or password results in database synchronization for both SNMP and AAA.

Cisco NX-OS synchronizes user configuration in the following ways:

- The **auth** passphrase specified in the **snmp-server user** command becomes the password for the CLI user.
- The password specified in the **username** command becomes the **auth** and **priv** passphrases for the SNMP user.
- If you create or delete a user using either SNMP or the CLI, the user is created or deleted for both SNMP and the CLI.
- User-role mapping changes are synchronized in SNMP and the CLI.
- Role changes (deletions or modifications from the CLI) are synchronized to SNMP.



Note When you configure passphrase/password in localized key/encrypted format, Cisco NX-OS does not synchronize the user information (passwords, rules, etc.).

Group-Based SNMP Access



Note Because a group is a standard SNMP term used industry-wide, roles are referred to as groups in this SNMP section.

SNMP access rights are organized by groups. Each group in SNMP is similar to a role through the CLI. Each group is defined with three accesses: read access, write access, and notification access. Each access can be enabled or disabled within each group.

You can begin communicating with the agent once your username is created, your roles are set up by your administrator, and you are added to the roles.

Guidelines and Limitations for SNMP

SNMP has the following configuration guidelines and limitations:

- Access control list (ACLs) can be applied only to local SNMPv3 users configured on the switch. ACLs cannot be applied to remote SNMPv3 users stored on Authentication, Authorization, and Accounting (AAA) servers.
- Cisco NX-OS supports read-only access to Ethernet MIBs. For more information, see the Cisco NX-OS MIB support list at the following URL <ftp://ftp.cisco.com/pub/mibs/supportlists/nexus3000/Nexus3000MIBSupportList.html>.
- Cisco NX-OS does not support the SNMPv3 `noAuthNoPriv` security level.
- Cisco Nexus 3600 series switches support up to 10000 flash files for `snmpwalk` request.
- Beginning with Cisco NX-OS Release 10.3(3)F, Type-6 encryption for SNMPv3 user password is supported with following limitations:
 - Type-6 encryption is successful only if the following is taken care:
 - **feature password encryption aes {tam}** is enabled.
 - Primary key is configured.
 - The **pwd_type 6** option is specified during SNMPv3 user configuration.
 - Changing the primary key configuration results in SNMP re-encrypting all Type-6 users stored in its database. However, the SNMP functionalities continue to work the same way as before.
 - Primary key configuration is local to the switch. If the user takes the Type-6 configured running data from one switch and applies it on other switch where a different primary key is configured, SNMP features for the same user might not work on the other switch.
 - If Type-6 is configured, ensure to remove the configuration, or reconfigure the Type-6 option before downgrading to the release where Type-6 is not supported.
 - In case of ISSU, if you migrate from an earlier image (where `localizedkey`, `localizedV2key` config is present) to a new image where Type-6 encryption is supported, SNMP won't convert the existing keys to Type-6 encryption.

- Conversion between existing SALT encryption to Type-6 encryption is supported using the **encryption re-encrypt obfuscated** command.
- ASCII-based reloads through disruptive upgrades and **reload-ascii** commands leads to loss of primary key which would impact the SNMP functionality for the Type-6 users.
- If a user enforces re-encryption using the **encryption re-encrypt obfuscated** command, then SNMP encrypts all passwords from non-Type-6 SNMP users to Type-6 mode.



Note The SNMP does not support the **encryption delete type6** command and a syslog warning message is also displayed indicating the same.

Default SNMP Settings

Table 15: Default SNMP Parameters

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

Configuring SNMP

Configuring the SNMP Source Interface

You can configure SNMP to use a specific interface.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **snmp-server source-interface** {inform | trap} *type slot/port*
3. switch(config)# **show snmp source-interface**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# snmp-server source-interface {inform trap} <i>type slot/port</i>	Configures the source interface for all SNMP packets. The following list contains the valid values for <i>interface</i> . <ul style="list-style-type: none"> • ethernet • loopback

	Command or Action	Purpose
		<ul style="list-style-type: none"> • mgmt • port-channel • vlan
Step 3	switch(config)# show snmp source-interface	Displays the configured SNMP source interface.

Example

This example shows how to configure the SNMP source interface:

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

Configuring SNMP Users



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

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server user** *name* [pwd_type 6] [auth {md5 | sha | sha-256 | sha-384 | sha-512} *passphrase* [auto] [priv [aes-128] *passphrase*] [engineID *id*] [localizedkey] | [localizedV2key]]
3. (Optional) switch# **show snmp user**
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	<p>snmp-server user <i>name</i> [pwd_type 6] [auth {md5 sha sha-256 sha-384 sha-512} <i>passphrase</i> [auto] [priv [aes-128] <i>passphrase</i>] [engineID id] [localizedkey] [localizedV2key]]</p> <p>Example:</p> <pre>switch(config)# snmp-server user Admin pwd_type 6 auth sha abcd1234 priv abcdefgh</pre>	<p>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.</p> <p>localizedkey - If you use the localizedkey keyword, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters. Instead of plain-text password, hashed password (copied either from the show running config command or generated offline using snmpv3 based open source hash generator tool, see Generating Hashed Password Offline, on page 117) can be configured using the localizedkey keyword.</p> <p>Note When using a localized key, add 0x before the hash value, for example,</p> <pre>0x84a716329158a97ac9f22780629bc26c.</pre> <p>localizedV2key - If the localizedV2key is used, the passphrase can be any case-sensitive, alphanumeric string up to 130 characters, without 0x at the beginning. Collect the localizedv2key using show run command, as this is an encrypted data and cannot be generated offline.</p> <p>The engineID format is a 12-digit, colon-separated decimal number.</p> <p>Note</p> <ul style="list-style-type: none"> Beginning with Cisco NX-OS Release 10.1(1), AES-128 is the default privacy protocol for SNMPv3. Beginning with Cisco NX-OS Release 10.3(3)F, the pwd_type 6 keyword is supported to provide Type-6 encryption for SNMP users password.
Step 3	<p>(Optional) switch# show snmp user</p> <p>Example:</p> <pre>switch(config) # show snmp user</pre>	Displays information about one or more SNMP users.
Step 4	<p>(Optional) copy running-config startup-config</p> <p>Example:</p> <pre>switch(config)# copy running-config startup-config</pre>	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

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
```

Generating Hashed Password Offline

Perform the following steps to generate hashed password offline, using snmpv3-based open source hash generator tool:



Note The IDs mentioned in this procedure are only sample IDs, the purpose of which is only to explain the procedure better.

1. Get the SNMP engineID from the switch.

```
switch# show snmp engineID
```

Sample output:

```
Local SNMP engineID: [Hex] 8000000903D4C93CEA31CC
[Dec] 128:000:000:009:003:212:201:060:234:049:204
```

2. Use an SNMPv3 based open source hash generator to generate offline hashed password.

```
Linux$ snmpv3-hashgen --auth Hello123 --engine 8000000903D4C93CEA31CC --user1 --mode priv
--hash md5
```

Sample output:

```
User: user1
Auth: Hello123 / 84a716329158a97ac9f22780629bc26c
Priv: Hello123 / 84a716329158a97ac9f22780629bc26c
Engine: 8000000903D4C93CEA31CC
ESXi USM String: u1/84a716329158a97ac9f22780629bc26c/84a716329158a97ac9f22780629bc26c/priv
```

3. Use the auth and priv values to configure the password on the switch.

```
snmp-server user user1 auth md5 0x84a716329158a97ac9f22780629bc26c priv des
0x84a716329158a97ac9f22780629bc26c localizedkey
```

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 <i>name</i> enforcePriv	Enforces SNMP message encryption for this user.

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

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

Assigning SNMPv3 Users to Multiple Roles

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



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

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

Creating SNMP Communities

You can create SNMP communities for SNMPv1 or SNMPv2c.

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

Filtering SNMP Requests

You can assign an access list (ACL) to a community to filter incoming SNMP requests. If the assigned ACL allows the incoming request packet, SNMP processes the request. If the ACL denies the request, SNMP drops the request and sends a system message.

Create the ACL with the following parameters:

- Source IP address
- Destination IP address
- Source port
- Destination port
- Protocol (UDP or TCP)

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



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

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

Command	Purpose
<pre>switch(config)# snmp-server community <i>community name</i> use-acl <i>acl-name</i></pre> <p>Example :</p> <pre>switch(config)# snmp-server community public use-acl my_acl_for_public</pre>	Assigns an IPv4 or IPv6 ACL to an SNMP community to filter SNMP requests.

Configuring SNMP Notification Receivers

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

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

Command	Purpose
<pre>switch(config)# snmp-server host <i>ip-address</i> traps version 1 <i>community</i> [<i>udp_port number</i>]</pre>	Configures a host receiver for SNMPv1 traps. The <i>ip-address</i> 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
<pre>switch(config)# snmp-server host <i>ip-address</i> {traps informs} version 2c <i>community</i> [<i>udp_port number</i>]</pre>	Configures a host receiver for SNMPv2c traps or informs. The <i>ip-address</i> 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
<pre>switch(config)# snmp-server host <i>ip-address</i> {traps informs} version 3 {auth noauth priv} <i>username</i> [<i>udp_port number</i>]</pre>	Configures a host receiver for SNMPv2c traps or informs. The <i>ip-address</i> 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

	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 the 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

	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 the 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

	Command or Action	Purpose
Step 1	switch# configuration terminal	Enters global configuration mode.
Step 2	switch(config)# snmp-server context <i>context-name</i> vrf <i>vrf-name</i>	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.
Step 3	switch(config)# snmp-server community <i>community-name</i> group <i>group-name</i>	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 <i>community-name</i> context <i>context-name</i>	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 16: Enabling SNMP Notifications

MIB	Related Commands
All notifications	snmp-server enable traps
CISCO-ERR-DISABLE-MIB	snmp-server enable traps show interface status
Q-BRIDGE-MIB	snmp-server enable traps show mac address-table
CISCO-SWITCH-QOS-MIB	snmp-server enable traps show hardware internal buffer info pkt-stats
BRIDGE-MIB	snmp-server enable traps bridge newroot snmp-server enable traps bridge topologychange
CISCO-AAA-SERVER-MIB	snmp-server enable traps aaa
ENTITY-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

MIB	Related Commands
CISCO-FSPF-MIB	snmp-server enable traps fspf
CISCO-PSM-MIB	snmp-server enable traps port-security
CISCO-RSCN-MIB	snmp-server enable traps rscn 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	snmp-server enable traps config
Note	Supports no MIB objects except the following notification: ccmCLIRunningConfigChanged



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

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	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: <pre>switch(config)# snmp-server enable traps link cieLinkDown</pre>	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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# interface <i>type slot/port</i>	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

	Command or Action	Purpose
Step 1	switch# configuration terminal	Enters global configuration mode.
Step 2	switch(config)# snmp-server contact <i>name</i>	Configures sysContact, the SNMP contact name.
Step 3	switch(config)# snmp-server location <i>name</i>	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

	Command or Action	Purpose
Step 1	switch# configuration terminal	Enters global configuration mode.
Step 2	switch(config)# snmp-server context <i>context-name</i> [instance <i>instance-name</i>] [vrf <i>vrf-name</i>] [topology <i>topology-name</i>]	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 <i>community-name</i> context <i>context-name</i>	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 <i>context-name</i> [instance <i>instance-name</i>] [vrf <i>vrf-name</i>] [topology <i>topology-name</i>]	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.

Configuring the SNMP Local Engine ID

Beginning with Cisco NX-OS Release 7.0(3)F3(1), you can configure the engine ID on a local device.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server engineID local** *engineid-string*
3. **show snmp engineID**
4. [**no**] **snmp-server engineID local** *engineid-string*
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	snmp-server engineID local <i>engineid-string</i> Example: switch(config)# snmp-server engineID local AA:BB:CC:1A:2C:10	Changes the SNMP engineID of the local device. The local engine ID should be configured as a list of colon-specified hexadecimal octets, where there are even number of hexadecimal characters that range from 10 to 64 and every two hexadecimal characters are separated by a colon. For example, i80:00:02:b8:04:61:62:63.
Step 3	show snmp engineID Example: switch(config)# show snmp engineID	Displays the identification of the configured SNMP engine.
Step 4	[no] snmp-server engineID local <i>engineid-string</i> Example: switch(config)# no snmp-server engineID local AA:BB:CC:1A:2C:10	Disables the local engine ID and the default auto-generated engine ID is configured.
Step 5	Required: copy running-config startup-config Example: switch(config)# copy running-config startup-config	Copies the running configuration to the startup configuration.

Disabling SNMP

SUMMARY STEPS

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

DETAILED STEPS

	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:	Disables SNMP. SNMP is disabled by default.

	Command or Action	Purpose
	no snmp-server protocol enable	

Verifying the SNMP Configuration

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

Command	Purpose
show snmp	Displays the SNMP status.
show snmp community	Displays the SNMP community strings.
show interface snmp-ifindex	Displays the SNMP ifIndex value for all interfaces (from IF-MIB).
show running-config snmp [all]	Displays the SNMP running configuration.
show snmp engineID	Displays the SNMP engineID.
show snmp group	Displays SNMP roles.
show snmp sessions	Displays SNMP sessions.
show snmp context	Displays the SNMP context mapping.
show snmp host	Displays information about configured SNMP hosts.
show snmp source-interface	Displays information about configured source interfaces.
show snmp trap	Displays the SNMP notifications enabled or disabled.
show snmp user	Displays SNMPv3 users.



CHAPTER 11

Using the PCAP SNMP Parser

This chapter contains the following sections:

- [Using the PCAP SNMP Parser, on page 131](#)

Using the PCAP SNMP Parser

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

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

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

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

Examples:

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

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

Examples:

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

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

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

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

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

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

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

Sessions
-----
1

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

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



CHAPTER 12

Configuring RMON

This chapter contains the following sections:

- [Information About RMON, on page 133](#)
- [Configuration Guidelines and Limitations for RMON, on page 134](#)
- [Verifying the RMON Configuration, on page 134](#)
- [Default RMON Settings, on page 135](#)
- [Configuring RMON Alarms, on page 135](#)
- [Configuring RMON Events, on page 136](#)

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 devices.

An RMON alarm monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified threshold value (threshold), and resets the alarm at another threshold value. You can use alarms with RMON events to generate a log entry or an SNMP notification when the RMON alarm triggers.

RMON is 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 the Cisco Nexus device triggers a falling alarm or resets a rising alarm.
- Events—The action that the Cisco Nexus device takes when an alarm (rising or falling) triggers.



Note Use the `hcalarms` option to set an alarm on a 64-bit integer MIB object.

For example, you can set a delta type rising alarm on an error counter MIB object. If the error counter delta exceeds this value, you can trigger an event that sends an SNMP notification and logs the rising alarm event. This rising alarm does not occur again until the delta sample for the error counter drops below the falling threshold.



Note The falling threshold must be less than the rising threshold.

RMON Events

You can associate a particular event to each RMON alarm. RMON supports the following event types:

- SNMP notification—Sends an SNMP risingAlarm or fallingAlarm notification when the associated alarm triggers.
- Log—Adds an entry in the RMON log table when the associated alarm triggers.
- Both—Sends an SNMP notification and adds an entry in the RMON log table when the associated alarm triggers.

You can specify a different even for a falling alarm and a rising alarm.

Configuration Guidelines and Limitations for RMON

RMON has the following configuration guidelines and limitations:

- You must configure an SNMP user and a notification receiver to use the SNMP notification event type.
- You can only configure an RMON alarm on a MIB object that resolves to an integer.

Verifying the RMON Configuration

Use the following commands to verify the RMON configuration information:

Command	Purpose
<code>show rmon alarms</code>	Displays information about RMON alarms.

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

Default RMON Settings

The following table lists the default settings for RMON parameters.

Table 17: Default RMON Parameters

Parameters	Default
Alarms	None configured.
Events	None configured.

Configuring RMON Alarms

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

You can optionally specify the following parameters:

- The eventnumber to trigger if the rising or falling threshold exceeds the specified limit.
- The owner of the alarm.

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

Before you begin

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

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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# rmon alarm <i>index mib-object sample-interval</i> {absolute delta} rising-threshold <i>value</i> [<i>event-index</i>] falling-threshold <i>value</i> [<i>event-index</i>] [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 <i>index mib-object sample-interval</i> {absolute delta} rising-threshold-high <i>value</i> rising-threshold-low <i>value</i> [<i>event-index</i>] falling-threshold-high <i>value</i> falling-threshold-low <i>value</i> [<i>event-index</i>] [owner name] [storagetype <i>type</i>]	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

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



CHAPTER 13

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 142](#)

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 3600 platform 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 18: 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 devices 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.

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 19: 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 20: Expansion Module Health Monitoring Diagnostics

Diagnostic	Description
LED	Monitors port and system status LEDs.

Diagnostic	Description
Temperature Sensor	Monitors temperature sensor readings.

Guidelines and Limitations for Online Diagnostics

Online diagnostics has the following configuration guidelines and limitations:

- You cannot run disruptive online diagnostic tests on demand.
- The BootupPortLoopback test is not supported.
- Interface Rx and Tx packet counters are incremented (approximately four packets every 15 minutes) for ports in the shutdown state.
- On admin down ports, the unicast packet Rx and Tx counters are incremented for GOLD loopback packets. The PortLoopback test is on demand, so the packet counter is incremented only when you run the test on admin down ports.

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

	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: <ul style="list-style-type: none"> • complete—Performs all bootup diagnostics. This is the default value. • bypass—Does not perform any bootup diagnostics.

	Command or Action	Purpose
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 21: Default Online Diagnostics Parameters

Parameters	Default
Bootup diagnostics level	complete



CHAPTER 14

Configuring the Embedded Event Manager

This chapter contains the following sections:

- [About Embedded Event Manager, on page 143](#)
- [Configuring Embedded Event Manager, on page 147](#)
- [Verifying the Embedded Event Manager Configuration, on page 174](#)
- [Configuration Examples for Embedded Event Manager, on page 175](#)
- [Additional References, on page 176](#)

About Embedded Event Manager

The ability to detect and handle critical events in the Cisco NX-OS system is important for high availability. The Embedded Event Manager (EEM) provides a central, policy-driven framework to detect and handle events in the system by monitoring events that occur on your device and taking action to recover or troubleshoot these events, based on your configuration..

EEM consists of three major components:

Event statements

Events to monitor from another Cisco NX-OS component that may require some action, workaround, or notification.

Action statements

An action that EEM can take, such as sending an e-mail or disabling an interface, to recover from an event.

Policies

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

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

Embedded Event Manager Policies

An EEM policy consists of an event statement and one or more action statements. The event statement defines the event to look for as well as the filtering characteristics for the event. The action statement defines the action EEM takes when the event occurs.

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

You can configure EEM policies using the command line interface (CLI) or a VSH script.

EEM gives you a device-wide view of policy management. Once EEM policies are configured, the corresponding actions are triggered. All actions (system or user-configured) for triggered events are tracked and maintained by the system.

Preconfigured System Policies

Cisco NX-OS has a number of preconfigured system policies. These system policies define many common events and actions for the device. System policy names begin with two underscore characters (___).

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



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

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

User-Created Policies

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

Log Files

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

Event Statements

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

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



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

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

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

Supported Events

EEM supports the following events in event statements:

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

Action Statements

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

In order for triggered events to process default actions, you must configure the EEM policy to allow the default action. For example, if you match a CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute.



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

Supported Actions

EEM supports the following actions in action statements:

- Execute any CLI commands
- Update a counter
- Reload the device
- Generate a syslog message
- Generate an SNMP notification

- Use the default action for the system policy

VSH Script Policies

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

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

Licensing Requirements for Embedded Event Manager

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

Prerequisites for Embedded Event Manager

You must have network-admin privileges to configure EEM.

Guidelines and Limitations for Embedded Event Manager

When you plan your EEM configuration, consider the following:

- The maximum number of configurable EEM policies is 500.
- Action statements within your user policy or overriding policy should not negate each other or adversely affect the associated system policy.
- 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.

Default Settings for Embedded Event Manager

Table 22: Default EEM Parameters

Parameters	Default
System Policies	Active

Configuring Embedded Event Manager

Defining an Environment Variable

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

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

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	event manager environment <i>variable-name variable-value</i> Example: switch(config) # event manager environment emailto "admin@anyplace.com"	Creates an environment variable for EEM. The <i>variable-name</i> can be any case-sensitive, alphanumeric string up to 29 characters. The <i>variable-value</i> can be any quoted case-sensitive, alphanumeric string up to 39 characters.
Step 3	(Optional) show event manager environment { <i>variable-name</i> 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.

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

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

	Command or Action	Purpose
Step 2	event manager applet <i>applet-name</i> Example: <pre>switch(config)# event manager applet monitorShutdown switch(config-applet)#</pre>	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 <i>policy-description</i> Example: <pre>switch(config-applet)# description "Monitors interface shutdown."</pre>	Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 4	event <i>event-statement</i> Example: <pre>switch(config-applet)# event cli match "shutdown"</pre>	Configures the event statement for the policy.
Step 5	(Optional) tag <i>tag</i> { and andnot or } <i>tag</i> [and andnot or { <i>tag</i> }] { happens <i>occurs in seconds</i> } Example: <pre>switch(config-applet)# tag one or two happens 1 in 10000</pre>	Correlates multiple events in the policy. The range for the <i>occurs</i> argument is from 1 to 4294967295. The range for the <i>seconds</i> argument is from 0 to 4294967295 seconds.
Step 6	action <i>number</i> [. <i>number2</i>] <i>action-statement</i> Example: <pre>switch(config-applet)# action 1.0 cli show interface e 3/1</pre>	Configures an action statement for the policy. Repeat this step for multiple action statements.
Step 7	(Optional) show event manager policy-state <i>name</i> [module <i>module-id</i>] Example: <pre>switch(config-applet)# show event manager policy-state monitorShutdown</pre>	Displays information about the status of the configured policy.
Step 8	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

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

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

	Command or Action	Purpose
	<code>switch(config-applet) # event fanbad time 3000</code>	The <i>number</i> range is module-dependent. The <i>seconds</i> range is from 10 to 64000.
Step 5	event memory {critical minor severe} Example: <code>switch(config-applet) # event memory critical</code>	Triggers an event if a memory threshold is crossed.
Step 6	event policy-default count repeats [time seconds] Example: <code>switch(config-applet) # event policy-default count 3</code>	Uses the event configured in the system policy. Use this option for overriding policies. The <i>repeats</i> range is from 1 to 65000. The <i>seconds</i> range is from 0 to 4294967295, where 0 indicates no time limit.
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: <code>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</code>	Triggers an event if the SNMP OID crosses the entry threshold based on the entry operation. The event resets immediately, or optionally you can configure the event to reset after the counter passes the exit threshold. The OID is in dotted decimal notation. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. The <i>entry</i> and <i>exit</i> value ranges are from 0 to 18446744073709551615. The <i>time</i> , in seconds, is from 0 to 2147483647. The <i>interval</i> , 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: <code>switch(config-applet) # event sysmgr memory minor 80</code>	Triggers an event if the specified system manager memory threshold is exceeded. The <i>percent</i> range is from 1 to 99.
Step 9	event temperature [module slot] [sensor number] threshold {any down up} Example: <code>switch(config-applet) # event temperature module 2 threshold any</code>	Triggers an event if the temperature sensor exceeds the configured threshold. The <i>sensor</i> range is from 1 to 18.
Step 10	event track [tag tag] object-number state {any down up} Example: <code>switch(config-applet) # event track 1 state down</code>	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 <i>object-number</i> 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

	Command or Action	Purpose
Step 1	action <i>number</i> [. <i>number2</i>] cli <i>command1</i> [<i>command2</i> .] [local] Example: <pre>switch(config-applet) # action 1.0 cli "show interface e 3/1"</pre>	Runs the configured commands. You can optionally run the commands on the module where the event occurred. The action label is in the format <i>number1.number2</i> . The <i>number</i> can be any number from 1 to 16 digits.

	Command or Action	Purpose
		The range for <i>number2</i> is from 0 to 9.
Step 2	<p>action <i>number</i>[.<i>number2</i>] counter name <i>counter</i> value <i>val</i> op {dec inc nop set}</p> <p>Example:</p> <pre>switch(config-applet) # action 2.0 counter name mycounter value 20 op inc</pre>	<p>Modifies the counter by the configured value and operation.</p> <p>The action label is in the format <i>number1.number2</i>.</p> <p>The <i>number</i> can be any number from 1 to 16 digits.</p> <p>The range for <i>number2</i> is from 0 to 9.</p> <p>The <i>counter</i> can be any case-sensitive, alphanumeric string up to 28 characters.</p> <p>The <i>val</i> can be an integer from 0 to 2147483647 or a substituted parameter.</p>
Step 3	<p>action <i>number</i>[.<i>number2</i>] event-default</p> <p>Example:</p> <pre>switch(config-applet) # action 1.0 event-default</pre>	<p>Completes the default action for the associated event.</p> <p>The action label is in the format <i>number1.number2</i>.</p> <p>The <i>number</i> can be any number from 1 to 16 digits.</p> <p>The range for <i>number2</i> is from 0 to 9.</p>
Step 4	<p>action <i>number</i>[.<i>number2</i>] policy-default</p> <p>Example:</p> <pre>switch(config-applet) # action 1.0 policy-default</pre>	<p>Completes the default action for the policy that you are overriding.</p> <p>The action label is in the format <i>number1.number2</i>.</p> <p>The <i>number</i> can be any number from 1 to 16 digits.</p> <p>The range for <i>number2</i> is from 0 to 9.</p>
Step 5	<p>action <i>number</i>[.<i>number2</i>] reload [module <i>slot</i> [- <i>slot</i>]]</p> <p>Example:</p> <pre>switch(config-applet) # action 1.0 reload module 3-5</pre>	<p>Forces one or more modules to the entire system to reload.</p> <p>The action label is in the format <i>number1.number2</i>.</p> <p>The <i>number</i> can be any number from 1 to 16 digits.</p> <p>The range for <i>number2</i> is from 0 to 9.</p>
Step 6	<p>action <i>number</i>[.<i>number2</i>] snmp-trap [intdata1 <i>integer-data1</i>] [intdata2 <i>integer-data2</i>] [strdata <i>string-data</i>]</p> <p>Example:</p> <pre>switch(config-applet) # action 1.0 snmp-trap strdata "temperature problem"</pre>	<p>Sends an SNMP trap with the configured data. The action label is in the format <i>number1.number2</i>.</p> <p>The <i>number</i> can be any number from 1 to 16 digits.</p> <p>The range for <i>number2</i> is from 0 to 9.</p> <p>The <i>data</i> elements can be any number up to 80 digits.</p> <p>The <i>string</i> can be any alphanumeric string up to 80 characters.</p>
Step 7	<p>action <i>number</i>[.<i>number2</i>] syslog [priority <i>prio-val</i>] msg <i>error-message</i></p> <p>Example:</p> <pre>switch(config-applet) # action 1.0 syslog priority notifications msg "cpu high"</pre>	<p>Sends a customized syslog message at the configured priority.</p> <p>The action label is in the format <i>number1.number2</i>.</p> <p>The <i>number</i> can be any number from 1 to 16 digits.</p> <p>The range for <i>number2</i> is from 0 to 9.</p>

	Command or Action	Purpose
		The <i>error-message</i> 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

-
- 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

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	event manager policy <i>policy-script</i> Example: <pre>switch(config)# event manager policy moduleScript</pre>	Registers and activates an EEM script policy. The <i>policy-script</i> can be any case-sensitive, alphanumeric string up to 29 characters.
Step 3	(Optional) event manager policy internal <i>name</i> Example: <pre>switch(config)# event manager policy internal moduleScript</pre>	Registers and activates an EEM script policy. The <i>policy-script</i> can be any case-sensitive alphanumeric string up to 29 characters.
Step 4	(Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

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

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	(Optional) show event manager policy-state system-policy Example: <pre>switch(config-applet)# show event manager policy-state __ethpm_link_flap Policy __ethpm_link_flap Cfg count : 5 Cfg time interval : 10.000000 (seconds) Hash default, Count 0</pre>	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.
Step 3	event manager applet applet-name override system-policy Example: <pre>switch(config-applet)# event manager applet ethport override __ethpm_link_flap switch(config-applet)#</pre>	Overrides a system policy and enters applet configuration mode. The <i>applet-name</i> can be any case-sensitive, alphanumeric string up to 80 characters. The <i>system-policy</i> must be one of the system policies.
Step 4	description policy-description Example: <pre>switch(config-applet)# description "Overrides link flap policy"</pre>	Configures a descriptive string for the policy. The <i>policy-description</i> 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: <pre>switch(config-applet)# event policy-default count 2 time 1000</pre>	Configures the event statement for the policy.
Step 6	section number action-statement Example: <pre>switch(config-applet)# action 1.0 syslog priority warnings msg "Link is flapping."</pre>	Configures an action statement for the policy. For multiple action statements, repeat this step.
Step 7	(Optional) show event manager policy-state name Example: <pre>switch(config-applet)# show event manager policy-state ethport</pre>	Displays information about the configured policy.

	Command or Action	Purpose
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

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	event manager applet <i>applet-name</i> 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 <i>tag</i>] { occurs <i>number</i> period <i>seconds</i> pattern <i>msg-text</i> priority <i>priority</i> }	Registers an applet with EEM and enters applet configuration mode.
	Example: switch(config-applet)# event syslog occurs 10	

	Command or Action	Purpose
Step 4	(Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code>	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to do next

Verify your EEM configuration.

Verifying the Embedded Event Manager Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show event manager environment [<i>variable-name</i> all]	Displays information about the event manager environment variables.
show event manager event-types [<i>event</i> all module <i>slot</i>]	Displays information about the event manager event types.
show event manager history events [detail] [maximum <i>num-events</i>] [severity { catastrophic minor moderate severe }]	Displays the history of events for all policies.
show event manager policy-state <i>policy-name</i>	Displays information about the policy state, including thresholds.
show event manager script system [<i>policy-name</i> all]	Displays information about the script policies.
show event manager system-policy [all]	Displays information about the predefined system policies.
show running-config eem	Displays information about the running configuration for EEM.
show startup-config eem	Displays information about the startup configuration for EEM.

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

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	bloggerd log-dump all Example: <pre>switch(config)# bloggerd log-dump all switch(config)#</pre>	Enables the log file retention feature for all services.

Example

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

Disabling Extended Log File Retention For All Services

Extended Log File Retention is 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

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	no bloggerd log-dump all Example: <pre>switch(config)# no bloggerd log-dump all switch(config)#</pre>	Disables the log file retention feature for all services on the switch.

Example

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

Enabling Extended Log File Retention For a Single Service

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

	Command or Action	Purpose
Step 1	show system internal sysmgr service name <i>service-type</i> Example: <pre>switch# show system internal sysmgr service name aclmgr</pre>	Displays information about the ACL Manager including the service SAP number.
Step 2	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	bloggerd log-dump sap <i>number</i> Example: <pre>switch(config)# bloggerd log-dump sap 351</pre>	Enables the log file retention feature for the ACL Manager service.
Step 4	show system internal bloggerd info log-dump-info Example: <pre>switch(config)# show system internal bloggerd info log-dump-info</pre>	Displays information about the log file retention feature on the switch.

Example

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
  UUID = 0x182, PID = 653, SAP = 351
  State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
  Restart count: 1
  Time of last restart: Mon Nov  4 11:10:39 2019.
  The service never crashed since the last reboot.
  Tag = N/A
  Plugin ID: 0

switch(config)# configure terminal
switch(config)# bloggerd log-dump sap 351
Sending Enable Request to Bloggerd
Bloggerd Log Dump Successfully enabled
switch(config)# show system internal bloggerd info log-dump-info
-----
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
-----
Module      | VDC      | SAP              | Enabled?
-----
1           | 1        | 351 (MTS_SAP_ACLMGR) | Enabled
-----

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

switch(config)#
```

Displaying Extended Log Files

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

SUMMARY STEPS

1. `dir debug:log-dump/`

DETAILED STEPS

	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

	Command or Action	Purpose
Step 1	show system internal sysmgr service name <i>service-type</i> 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 <i>number</i> Example: switch(config)# no bloggerd log-dump sap 351	Disables the log file retention feature for the ACL Manager service.

	Command or Action	Purpose
Step 4	<p>show system internal bloggerd info log-dump-info</p> <p>Example:</p> <pre>switch(config)# show system internal bloggerd info log-dump-info</pre>	Displays information about the log file retention feature on the switch.

Example

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

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
  UUID = 0x182, PID = 653, SAP = 351
  State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
  Restart count: 1
  Time of last restart: Mon Nov  4 11:10:39 2019.
  The service never crashed since the last reboot.
  Tag = N/A
  Plugin ID: 0
switch(config)# configure terminal
switch(config)# no bloggerd log-dump sap 351
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)# show system internal bloggerd info log-dump-info
-----
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
-----
Module      | VDC      | SAP              | Enabled?
-----
1           | 1        | 351 (MTS_SAP_ACLMGR) | Disabled
-----
Log Dump Throttle Switch-Wide Config:
-----
Log Dump Throttle                               : ENABLED
Minimum buffer rollover count (before throttling) : 5
Maximum allowed rollover count per minute        : 1
-----

switch(config)#
```

Trigger-Based Event Log Auto-Collection

Trigger-based log collection capabilities:

- Automatically collect relevant data when issues occur.
- No impact on control plane
- Customizable configuration:
 - Defaults populated by Cisco
 - Selectively override what-to-collect by network administrator or by Cisco TAC.

- Automatically update new triggers on image upgrades.
- Store logs locally on the switch or remotely on an external server.
- Supports severity 0, 1, and 2 syslogs:
- Custom syslogs for ad-hoc events (auto-collection commands attached to the syslogs)

Enabling Trigger-Based Log File Auto-Collection

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

For more information on creating a custom YAML file to enable log file auto-collection, see [Configuring the Auto-Collection YAML File, on page 164](#).

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 (<code>securityd</code> is a facility name in syslog).
default:	Identifies all messages belonging to the component.
tech-sup: port	Collect tech support of the port module for the <code>securityd</code> syslog component.
commands: show module	Collect show module command output for the <code>securityd</code> syslog component.
platform:	Name of the syslog component (<code>platform</code> is a facility name in syslog).
tech-sup: port	Collect tech support of the port module for the <code>platform</code> syslog component.
commands: show module	Collect show module command output for the <code>platform</code> syslog component.

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

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

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

Example syslog output for the above YAML entry:

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

Use the following example to specify multiple values.

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



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

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

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

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

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

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

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

Limiting the Amount of Auto-Collections Per Component

For auto-collection, the limit of the number of bundles per component event is set to 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 <code>bloggerd log-snapshot <></code> command.

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

```

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

```

Parsing the Log tar Files

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

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

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

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

Copying Logs to a Different Location

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

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

Purging Auto-Collection Log Files

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

Purge Logic for EventHistory Logs

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

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

Purge Logic for EventBundle Logs

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

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

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

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

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

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

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



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

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

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

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

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

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

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



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

Auto-Collection Statistics and History

The following example shows trigger-based collection statistics:

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

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

```
switch# show system internal event-logs auto-collect history
DateTime Snapshot ID Syslog Status/Secs/Logsize(Bytes)
2019-Dec-04 05:30:32 1310232084 VPC-0-TEST_SYSLOG PROCESSED:9:22312929
2019-Dec-04 05:30:22 1310232084 VPC-0-TEST_SYSLOG PROCESSING
2019-Dec-04 04:30:13 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSED:173:33194665
2019-Dec-04 04:28:47 897805674 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:28:47 947981421 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:27:19 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSING
2019-Dec-04 02:17:16 1957148102 CARDCLIENT-2-FPGA_BOOT_GOLDEN 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 160.
 - Export the internal event logs off the switch. See [External Log File Storage](#), on page 173.
- 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

	Command or Action	Purpose
Step 1	<p>bloggerd log-snapshot [<i>file-name</i>] [bootflash: <i>file-path</i> logflash: <i>file-path</i> usb1:] [size <i>file-size</i>] [time <i>minutes</i>]</p> <p>Example:</p> <pre>switch# bloggerd log-snapshot snapshot1</pre>	<p>Creates a snapshot bundle file of the last ten event logs stored on the switch. Default storage for this operation is logflash.</p> <p><i>file-name</i>: The filename of the generated snapshot log file bundle. Use a maximum of 64 characters for <i>file-name</i>.</p> <p>Note This variable is optional. If it is not configured, the system applies a timestamp and "_snapshot_bundle.tar" as the filename. Example:</p> <pre>20200605161704_snapshot_bundle.tar</pre>

	Command or Action	Purpose
		<p>bootflash: <i>file-path</i>: The file path where the snapshot log file bundle is being stored on the bootflash. Choose one of the following initial paths:</p> <ul style="list-style-type: none"> • bootflash:/// • bootflash://module-1/ • bootflash://sup-1/ • bootflash://sup-active/ • bootflash://sup-local/ <p>logflash: <i>file-path</i>: The file path where the snapshot log file bundle is being stored on the logflash. Choose one of the following initial paths:</p> <ul style="list-style-type: none"> • logflash:/// • logflash://module-1/ • logflash://sup-1/ • logflash://sup-active/ • logflash://sup-local/ <p>usb1: The file path where the snapshot log file bundle is being stored on the USB device.</p> <p>size <i>file-size</i>: The snapshot log file bundle based on size in megabytes (MB). Range is from 5MB through 250MB.</p> <p>time <i>minutes</i>: The snapshot log file bundle based on the last x amount of time (minutes). Range is from 1 minute through 30 minutes.</p>

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

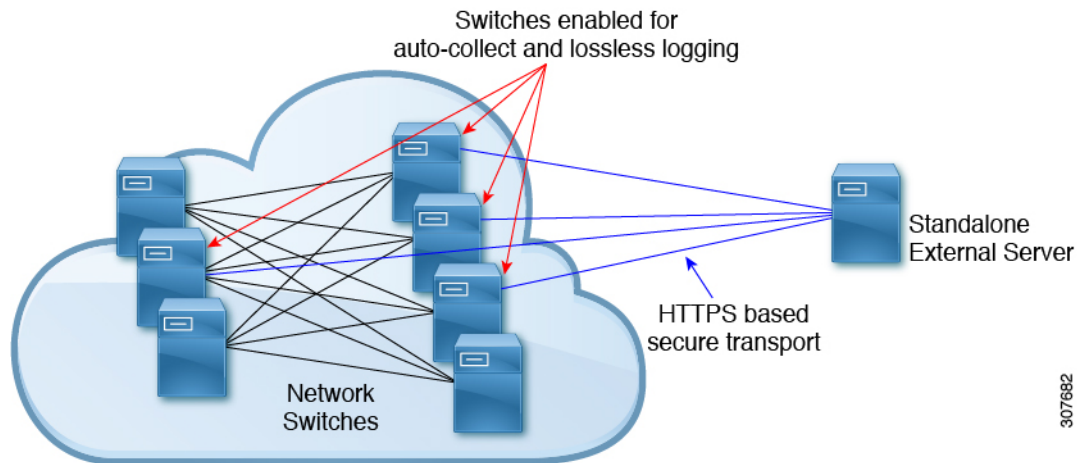
An external server solution provides the capability to store logs off-switch in a secure manner.



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

The following are external log file storage capabilities:

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



307682

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.

Verifying the Embedded Event Manager Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show event manager environment [<i>variable-name</i> all]	Displays information about the event manager environment variables.
show event manager event-types [<i>event</i> all module slot]	Displays information about the event manager event types.
show event manager history events [detail] [maximum num-events] [severity {catastrophic minor moderate severe}]	Displays the history of events for all policies.
show event manager policy-state <i>policy-name</i>	Displays information about the policy state, including thresholds.

Command	Purpose
<code>show event manager script system [policy-name all]</code>	Displays information about the script policies.
<code>show event manager system-policy [all]</code>	Displays information about the predefined system policies.
<code>show running-config eem</code>	Displays information about the running configuration for EEM.
<code>show startup-config eem</code>	Displays information about the startup configuration for EEM.

Configuration Examples for Embedded Event Manager

The following example shows how to override the `__lcm_module_failure` system policy by changing the threshold for only module 3 hitless upgrade failures. It also sends a syslog message. The settings in the system policy, `__lcm_module_failure`, apply in all other cases.

```
event manager applet example2 override __lcm_module_failure
event module-failure type hitless-upgrade-failure module 3 count 2
  action 1 syslog priority errors msg module 3 "upgrade is not a hitless upgrade!"
  action 2 policy-default
```

The following example shows how to override the `__ethpm_link_flap` system policy and shut down the interface:

```
event manager applet ethport override __ethpm_link_flap
  event policy-default count 2 time 1000
  action 1 cli conf t
  action 2 cli int et1/1
  action 3 cli no shut
```

The following example shows how to create an EEM policy that allows the command to execute but triggers an SNMP notification when a user enters configuration mode on the device:

```
event manager applet TEST
  event cli match "conf t"
  action 1.0 snmp-trap strdata "Configuration change"
  action 2.0 event-default
```



Note You must add the **event-default** action statement to the EEM policy or EEM does not allow the command to execute.

The following example shows how to correlate multiple events in an EEM policy and execute the policy based on a combination of the event triggers. In this example, the EEM policy is triggered if one of the specified syslog patterns occurs within 120 seconds.

```
event manager applet eem-correlate
  event syslog tag one pattern "copy bootflash:.* running-config.*"
  event syslog tag two pattern "copy run start"
  event syslog tag three pattern "hello"
  tag one or two or three happens 1 in 120
  action 1.0 reload module 1
```

Additional References

Related Documents

Related Topic	Document Title
EEM commands	<i>Cisco Nexus 3600 NX-OS Command Reference</i>

Standards

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



CHAPTER 15

Configuring Onboard Failure Logging

This chapter contains the following sections:

- [About OBFL, on page 177](#)
- [Prerequisites for OBFL, on page 178](#)
- [Guidelines and Limitations for OBFL, on page 178](#)
- [Default Settings for OBFL, on page 178](#)
- [Configuring OBFL, on page 178](#)
- [Verifying the OBFL Configuration, on page 181](#)
- [Configuration Example for OBFL, on page 182](#)
- [Additional References, on page 182](#)

About OBFL

Cisco NX-OS provides the ability to log failure data to persistent storage, which you can retrieve and display for analysis at a later time. This onboard failure logging (OBFL) feature stores failure and environmental information in nonvolatile memory on the module. The information will help analyze failed modules.

OBFL stores the following types of data:

- Time of initial power-on
- Slot number of the module in the chassis
- Initial temperature of the module
- Firmware, BIOS, FPGA, and ASIC versions
- Serial number of the module
- Stack trace for crashes
- CPU hog information
- Memory leak information
- Software error messages
- Hardware exception logs
- Environmental history

- OBFL-specific history information
- ASIC interrupt and error statistics history
- ASIC register dumps

Prerequisites for OBFL

You must have network-admin user privileges.

Guidelines and Limitations for OBFL

OBFL has the following guidelines and limitations:

- OBFL is enabled by default.
- OBFL flash supports a limited number of writes and erases. The more logging you enable, the faster you use up this number of writes and erases.
- The **show system reset-reason module *module num*** command does not display the reset reason in case of a module failure. Due to lack of persistent storage of the module reset-reason, this command is not effective after a reboot. Since the exception log is available in persistent storage, after a reboot, you can view the reset-reason using the **show logging onboard exception-log** command.



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

Default Settings for OBFL

The following table lists the default settings for OBFL parameters.

Parameters	Default
OBFL	All features enabled

Configuring OBFL

You can configure the OBFL features on Cisco NX-OS devices.

Before you begin

Make sure that you are in global configuration mode.

SUMMARY STEPS

1. **configure terminal**
2. **hw-module logging onboard**
3. **hw-module logging onboard counter-stats**
4. **hw-module logging onboard cpuhog**
5. **hw-module logging onboard environmental-history**
6. **hw-module logging onboard error-stats**
7. **hw-module logging onboard interrupt-stats**
8. **hw-module logging onboard module *slot***
9. **hw-module logging onboard obfl-logs**
10. (Optional) **show logging onboard**
11. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	hw-module logging onboard Example: <pre>switch(config)# hw-module logging onboard Module: 7 Enabling ... was successful. Module: 10 Enabling ... was successful. Module: 12 Enabling ... was successful.</pre>	Enables all OBFL features.
Step 3	hw-module logging onboard counter-stats Example: <pre>switch(config)# hw-module logging onboard counter-stats Module: 7 Enabling counter-stats ... was successful. Module: 10 Enabling counter-stats ... was successful. Module: 12 Enabling counter-stats ... was successful.</pre>	Enables the OBFL counter statistics.
Step 4	hw-module logging onboard cpuhog Example: <pre>switch(config)# hw-module logging onboard cpuhog Module: 7 Enabling cpu-hog ... was successful. Module: 10 Enabling cpu-hog ... was successful. Module: 12 Enabling cpu-hog ... was successful.</pre>	Enables the OBFL CPU hog events.
Step 5	hw-module logging onboard environmental-history Example:	Enables the OBFL environmental history.

	Command or Action	Purpose
	<pre>switch(config)# hw-module logging onboard environmental-history Module: 7 Enabling environmental-history ... was successful. Module: 10 Enabling environmental-history ... was successful. Module: 12 Enabling environmental-history ... was successful.</pre>	
Step 6	<p>hw-module logging onboard error-stats</p> <p>Example:</p> <pre>switch(config)# hw-module logging onboard error-stats Module: 7 Enabling error-stats ... was successful. Module: 10 Enabling error-stats ... was successful. Module: 12 Enabling error-stats ... was successful.</pre>	Enables the OBFL error statistics.
Step 7	<p>hw-module logging onboard interrupt-stats</p> <p>Example:</p> <pre>switch(config)# hw-module logging onboard interrupt-stats Module: 7 Enabling interrupt-stats ... was successful. Module: 10 Enabling interrupt-stats ... was successful. Module: 12 Enabling interrupt-stats ... was successful.</pre>	Enables the OBFL interrupt statistics.
Step 8	<p>hw-module logging onboard module <i>slot</i></p> <p>Example:</p> <pre>switch(config)# hw-module logging onboard module 7 Module: 7 Enabling ... was successful.</pre>	Enables the OBFL information for a module.
Step 9	<p>hw-module logging onboard obfl-logs</p> <p>Example:</p> <pre>switch(config)# hw-module logging onboard obfl-logs Module: 7 Enabling obfl-log ... was successful. Module: 10 Enabling obfl-log ... was successful. Module: 12 Enabling obfl-log ... was successful.</pre>	Enables the boot uptime, device version, and OBFL history.
Step 10	<p>(Optional) show logging onboard</p> <p>Example:</p> <pre>switch(config)# show logging onboard</pre>	Displays information about OBFL.
Step 11	<p>(Optional) copy running-config startup-config</p> <p>Example:</p> <pre>switch(config)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Verifying the OBFL Configuration

To display OBFL information stored in flash on a module, perform one of the following tasks:

Command	Purpose
show logging onboard boot-uptime	Displays the boot and uptime information.
show logging onboard counter-stats	Displays statistics on all ASIC counters.
show logging onboard credit-loss	Displays OBFL credit loss logs.
show logging onboard device-version	Displays device version information.
show logging onboard endtime	Displays OBFL logs to a specified end time.
show logging onboard environmental-history	Displays environmental history.
show logging onboard error-stats	Displays error statistics.
show logging onboard exception-log	Displays exception log information.
show logging onboard interrupt-stats	Displays interrupt statistics.
show logging onboard module <i>slot</i>	Displays OBFL information for a specific module.
show logging onboard obfl-history	Displays history information.
show logging onboard obfl-logs	Displays log information.
show logging onboard stack-trace	Displays kernel stack trace information.
show logging onboard starttime	Displays OBFL logs from a specified start time.
show logging onboard status	Displays OBFL status information.

Use the **show logging onboard status** command to display the configuration status of OBFL.

```
switch# show logging onboard status
-----
OBFL Status
-----
Switch OBFL Log: Enabled

Module: 4 OBFL Log: Enabled
cpu-hog Enabled
credit-loss Enabled
environmental-history Enabled
error-stats Enabled
exception-log Enabled
interrupt-stats Enabled
mem-leak Enabled
miscellaneous-error Enabled
obfl-log (boot-uptime/device-version/obfl-history) Enabled
register-log Enabled
request-timeout Enabled
stack-trace Enabled
system-health Enabled
timeout-drops Enabled
stack-trace Enabled

Module: 22 OBFL Log: Enabled
```

```

cpu-hog Enabled
credit-loss Enabled
environmental-history Enabled
error-stats Enabled
exception-log Enabled
interrupt-stats Enabled
mem-leak Enabled
miscellaneous-error Enabled
obfl-log (boot-uptime/device-version/obfl-history) Enabled
register-log Enabled
request-timeout Enabled
stack-trace Enabled
system-health Enabled
timeout-drops Enabled
stack-trace Enabled

```

Use the **clear logging onboard** command to clear the OBFL information for each of the **show** command options listed.

Configuration Example for OBFL

This example shows how to enable OBFL on module 2 for environmental information:

```

switch# configure terminal
switch(config)# hw-module logging onboard module 2 environmental-history

```

Additional References

Related Documents

Related Topic	Document Title
Configuration files	<i>Cisco Nexus 3600 NX-OS Fundamentals Configuration Guide</i>



CHAPTER 16

Configuring SPAN

This chapter contains the following sections:

- [Information About SPAN, on page 183](#)
- [SPAN Sources, on page 183](#)
- [Characteristics of Source Ports, on page 184](#)
- [SPAN Destinations, on page 184](#)
- [Characteristics of Destination Ports, on page 184](#)
- [Guidelines and Limitations for SPAN, on page 185](#)
- [Creating or Deleting a SPAN Session, on page 186](#)
- [Configuring an Ethernet Destination Port, on page 186](#)
- [Configuring Source Ports, on page 188](#)
- [Configuring the Rate Limit for SPAN Traffic, on page 188](#)
- [Configuring Source Port Channels or VLANs, on page 189](#)
- [Configuring the Description of a SPAN Session, on page 190](#)
- [Activating a SPAN Session, on page 191](#)
- [Suspending a SPAN Session, on page 191](#)
- [Displaying SPAN Information, on page 192](#)
- [Configuration Examples for SPAN, 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, a Fibre Channel Analyzer, or other Remote Monitoring (RMON) probes.

The Switched Port Analyzer (SPAN) feature (sometimes called port mirroring or port monitoring) selects network traffic for analysis by a network analyzer. The network analyzer can be a Cisco SwitchProbe or other Remote Monitoring (RMON) probes.

SPAN Sources

SPAN sources refer to the interfaces from which traffic can be monitored. The Cisco Nexus device supports Ethernet, 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.
- SPAN sources for VLANs cannot be more than 6 VLANs.
- Without an ACL filter configured, the same source can be configured for multiple sessions as long as either the direction or SPAN destination is different. However, each SPAN RX source should be configured for only one SPAN session with an ACL filter.
- Cannot be a destination port.
- Can be configured with a direction (ingress, egress, or both) to monitor. For VLAN sources, the monitored direction can only be ingress and applies to all physical ports in the group. The RX/TX option is not available for VLAN SPAN sessions.
- Ingress traffic can be filtered by using ACLs so that they mirror only those packets of information that match the ACL criteria.
- Can be in the same or different VLANs.

SPAN Destinations

SPAN destinations refer to the interfaces that monitors source ports. The Cisco Nexus 3600 platform switches support 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, FCoE, and Fibre Channel ports cannot be destination ports.
- 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.

Guidelines and Limitations for SPAN



Note For scale information, see the release-specific *Cisco Nexus 3600 NX-OS Verified Scalability Guide*.

SPAN has the following guidelines and limitations:

- The same source (ethernet or port-channel) can be a part of multiple sessions. You can configure two monitor session with different destinations, but the same source VLAN is not supported.
- Multiple ACL filters are supported on the same source.
- An egress SPAN copy of an access port on Cisco Nexus 3600 platform switch interfaces will always have a dot1q header.
- 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.
- ACL filtering is supported only for Rx SPAN. Tx SPAN mirrors all traffics that egresses at the source interface.
- TCAM carving is not required for SPAN/ERSPAN on Cisco Nexus 3600 platform switches.
- ACL filtering is not supported for IPv6 and MAC ACLs because of ternary content addressable memory (TCAM) width limitations.
- The SPAN TCAM size is 128 or 256, depending on the ASIC. One entry is installed as the default and four are reserved for ERSPAN.
- If the same source is configured in more than one SPAN session, and each session has an ACL filter configured, the source interface is programmed only for the first active SPAN session. Hardware entries programmed for ACEs in other sessions is not included in this source interface.
- Both permit and deny access control entries (ACEs) are treated alike. Packets that match the ACE are mirrored irrespective of whether they have a permit or deny entry in the ACL.



Note A deny ACE does not result in a dropped packet. An ACL configured in a SPAN session determines only whether the packet is mirrored or not.

- It is recommended to use only the RX type of source traffic for SPAN to provide better performance because RX traffic is cut-through, whereas TX is store-and-forward. Hence, when monitoring both directions (RX and TX), the performance is not as good as when monitoring only RX. If you need to monitor both directions of traffic, you can monitor RX on more physical ports to capture both sides of the traffic.
- Beginning with Cisco NX-OS Release 10.2(3)F, ACL filters are supported on following platform switches:
 - N3K-C36180YC-R
 - N3K-C3636C-R

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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# monitor session <i>session-number</i>	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

	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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # monitor session <i>session-number</i>	Enters monitor configuration mode for the specified monitoring session.
Step 3	switch(config-monitor) # source interface <i>type slot/port</i> [rx tx both]	Adds an Ethernet SPAN source port and specifies the traffic direction in which to duplicate packets. You can enter a range of Ethernet, Fibre Channel, or virtual Fibre Channel ports. You can specify the traffic direction to duplicate as ingress (Rx), egress (Tx), or both. By default, the direction is both.

Example

The following example shows how to configure an Ethernet SPAN source port:

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# source interface ethernet 1/16
switch(config-monitor)#
```

The following example shows how to configure a Fibre Channel SPAN source port:

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# source interface fc 2/1
switch(config-monitor)#
```

Configuring the Rate Limit for SPAN Traffic

By configuring a rate limit for SPAN traffic to 1Gbps across the entire monitor session, you can avoid impacting the monitored production traffic.

- When spanning more than 1Gbps to a 1 Gb SPAN destination interface, SPAN source traffic will not drop.
- When spanning more than 6 Gbps (but less than 10Gbps) to a 10Gb SPAN destination interface, the SPAN traffic is limited to 1Gbps even though the destination/sniffer is capable of 10Gbps.
- SPAN is rate-limited to 5 Gbps for every 8 ports (one ASIC).
- RX-SPAN is rate-limited to 0.71 Gbps per port when the RX-traffic on the port exceeds 5 Gbps.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface ethernet slot/port**
3. switch(config-if)# **switchport monitor rate-limit 1G**
4. switch(config-if)# **exit**

DETAILED STEPS

	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 specified Ethernet interface selected by the slot and port values. Note If this is a QSFP+ GEM, the <i>slot/port</i> syntax is <i>slot/QSFP-module/port</i> .
Step 3	switch(config-if)# switchport monitor rate-limit 1G	Specifies that the rate limit is 1 Gbps. Note This command is not supported on the Cisco Nexus N3K-C36180YC-R platform switch.
Step 4	switch(config-if)# exit	Reverts to global configuration mode.

Example

This example shows how to limit the bandwidth on Ethernet interface 1/2 to 1 Gbps:

```
switch(config)# interface ethernet 1/2
switch(config-if)# switchport monitor rate-limit 1G
switch(config-if)#
```

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**} *channel-number* [**rx** | **tx** | **both**] | **vlan** *vlan-range*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # monitor session <i>session-number</i>	Enters monitor configuration mode for the specified SPAN session.
Step 3	switch(config-monitor) # source { interface { port-channel } <i>channel-number</i> [rx tx both] vlan <i>vlan-range</i> }	Configures port channel or VLAN sources. For VLAN sources, the monitored direction is implicit.

Example

The following example shows how to configure a port channel SPAN source:

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

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

	Command or Action	Purpose
Step 2	switch(config) # monitor session <i>session-number</i>	Enters monitor configuration mode for the specified SPAN session.
Step 3	switch(config-monitor) # description <i>description</i>	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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # no monitor session {all <i>session-number</i> } 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**

- switch(config) # **monitor session** {all | *session-number*} **shut**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # monitor session {all <i>session-number</i> } shut	Suspends the specified SPAN session or all sessions.

Example

The following example shows how to suspend a SPAN session:

```
switch# configure terminal
switch(config) # monitor session 3 shut
switch(config) #
```

Displaying SPAN Information

SUMMARY STEPS

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

DETAILED STEPS

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

Example

The following example shows how to display SPAN session information:

```
switch# show monitor
SESSION STATE REASON DESCRIPTION
-----
2 up The session is up
3 down Session suspended
4 down No hardware resource
```

The following example shows how to display SPAN session details:

```
switch# show monitor session 2
session 2
-----
type : local
state : up
source intf :

source VLANs :
rx : 100
```

```
tx          :  
both       :  
destination ports : Eth3/1
```

Configuration Examples for SPAN

Configuration Example for a SPAN Session

To configure a SPAN session, follow these steps:

SUMMARY STEPS

1. Configure destination ports in access mode and enable SPAN monitoring.
2. Configure a SPAN session.

DETAILED STEPS

Step 1 Configure destination ports in access mode and enable SPAN monitoring.

Example:

```
switch# configure terminal  
switch(config)# interface ethernet 2/5  
switch(config-if)# switchport  
switch(config-if)# switchport monitor  
switch(config-if)# no shut  
switch(config-if)# exit  
switch(config)#
```

Step 2 Configure a SPAN session.

Example:

```
switch(config)# no monitor session 3  
switch(config)# monitor session 3  
switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx  
switch(config-monitor)# source interface port-channel 2  
switch(config-monitor)# source interface sup-eth 0 both  
switch(config-monitor)# source vlan 3, 6-8 rx  
switch(config-monitor)# source interface ethernet 101/1/1-3  
switch(config-monitor)# destination interface ethernet 2/5  
switch(config-monitor)# no shut  
switch(config-monitor)# exit  
switch(config)# show monitor session 3  
switch(config)# copy running-config startup-config
```

Configuration Example for a Unidirectional SPAN Session

To configure a unidirectional SPAN session, follow these steps:

SUMMARY STEPS

1. Configure destination ports in access mode and enable SPAN monitoring.
2. Configure a SPAN session.

DETAILED STEPS

Step 1 Configure destination ports in access mode and enable SPAN monitoring.

Example:

```
switch# configure terminal
switch(config)# interface ethernet 2/5
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# no shut
switch(config-if)# exit
switch(config)#
```

Step 2 Configure a SPAN session.

Example:

```
switch(config)# no monitor session 3
switch(config)# monitor session 3 rx
switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx
switch(config-monitor)# filter vlan 3-5, 7
switch(config-monitor)# destination interface ethernet 2/5
switch(config-monitor)# no shut
switch(config-monitor)# exit
switch(config)# show monitor session 3
switch(config)# copy running-config startup-config
```

Configuration Example for a SPAN ACL

This example shows how to configure a SPAN ACL:

```
switch# configure terminal
switch(config)# ip access-list match_11_pkts
switch(config-acl)# permit ip 11.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# ip access-list match_12_pkts
switch(config-acl)# permit ip 12.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# vlan access-map span_filter 5
switch(config-access-map)# match ip address match_11_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# vlan access-map span_filter 10
switch(config-access-map)# match ip address match_12_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# monitor session 1
switch(config-erspan-src)# filter access-group span_filter
```

Configuration Examples for UDF-Based SPAN

This example shows how to configure UDF-based SPAN to match on the inner TCP flags of an encapsulated IP-in-IP packet using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + Outer IP (20) + Inner IP (20) + Inner TCP (20, but TCP flags at 13th byte)
- Offset from packet-start: 14 + 20 + 20 + 13 = 67
- UDF match value: 0x20
- UDF mask: 0xFF

```

udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify udf udf_tcpflags
copy running-config startup-config
reload
ip access-list acl-udf
  permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1
  source interface Ethernet 1/1
  filter access-group acl-udf

```

This example shows how to configure UDF-based SPAN to match regular IP packets with a packet signature (DEADBEEF) at 6 bytes after a Layer 4 header start using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + IP (20) + TCP (20) + Payload: 112233445566DEADBEEF7788
- Offset from Layer 4 header start: 20 + 6 = 26
- UDF match value: 0xDEADBEEF (split into two-byte chunks and two UDFs)
- UDF mask: 0xFFFFFFFF

```

udf udf_pktsig_msb header outer 14 26 2
udf udf_pktsig_lsb header outer 14 28 2
hardware access-list tcam region racl qualify udf udf_pktsig_msb udf_pktsig_lsb
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
  permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1
  source interface Ethernet 1/1
  filter access-group acl-udf-pktsig

```




CHAPTER 17

Configuring ERSPAN

This chapter contains the following sections:

- [About ERSPAN, on page 197](#)
- [Prerequisites for ERSPAN, on page 198](#)
- [Guidelines and Limitations for ERSPAN, on page 198](#)
- [Default Settings for ERSPAN, on page 201](#)
- [Configuring ERSPAN, on page 201](#)
- [Configuration Examples for ERSPAN, on page 214](#)
- [Additional References, on page 216](#)

About ERSPAN

ERSPAN consists of an ERSPAN source session, routable ERSPAN generic routing encapsulation (GRE)-encapsulated traffic, and an ERSPAN destination session. You can separately configure ERSPAN source sessions and destination sessions on different switches. You can also configure ERSPAN source sessions to filter ingress traffic by using ACLs.

ERSPAN Sources

The interfaces from which traffic can be monitored are called ERSPAN sources. Sources designate the traffic to monitor and whether to copy ingress, egress, or both directions of traffic. ERSPAN sources include the following:

- Ethernet ports, port channels, and subinterfaces.
- VLANs—When a VLAN is specified as an ERSPAN source, all supported interfaces in the VLAN are ERSPAN sources.

ERSPAN source ports have the following characteristics:

- A port configured as a source port cannot also be configured as a destination port.
- ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.
- Ingress traffic at source ports can be filtered by using ACLs so that they mirror only those packets of information that match the ACL criteria.

Multiple ERSPAN Sessions

Although you can define up to 18 ERSPAN sessions, only a maximum of four ERSPAN or SPAN sessions can be operational simultaneously. If both receive and transmit sources are configured in the same session, only two ERSPAN or SPAN sessions can be operational simultaneously. You can shut down any unused ERSPAN sessions.

For information about shutting down ERSPAN sessions, see [Shutting Down or Activating an ERSPAN Session, on page 212](#).

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



Note For scale information, see the release-specific *Cisco Nexus 3600 NX-OS Verified Scalability Guide*.

ERSPAN has the following configuration guidelines and limitations:

- The same source can be part of multiple sessions.
- Multiple ACL filters are supported on the same source.
- ERSPAN supports the following:
 - From 4 to 6 tunnels
 - Nontunnel packets
 - IPinIP tunnels
 - IPv4 tunnels (limited)
 - ERSPAN source session type (packets are encapsulated as generic routing encapsulation (GRE)-tunnel packets and sent on the IP network. However, unlike other Cisco devices, the ERSPAN header is not added to the packet.)
- ERSPAN packets are dropped if the encapsulated mirror packet fails Layer 2 MTU checks.
- There is a 112-byte limit for egress encapsulation. Packets that exceed this limit are dropped. This scenario might be encountered when tunnels and mirroring are intermixed.

- ERSPAN sessions are shared with local sessions. A maximum of 18 sessions can be configured; however only a maximum of four sessions can be operational at the same time. If both receive and transmit sources are configured in the same session, only two sessions can be operational.
- ERSPAN and ERSPAN ACLs are not supported for packets that are generated by the supervisor.
- ERSPAN and ERSPAN with ACL filtering are not supported for packets that are generated by the supervisor.
- ACL filtering is supported only for Rx ERSPAN. Tx ERSPAN that mirrors all traffic that is egressed at the source interface.
- ACL filtering is not supported for IPv6 and MAC ACLs because of TCAM width limitations.
- If the same source is configured in more than one ERSPAN session, and each session has an ACL filter that is configured, the source interface is programmed only for the first active ERSPAN session. The ACEs that belong to the other sessions will not have this source interface programmed.
- If you configure an ERSPAN session and a local SPAN session (with filter access-group and allow-sharing option) to use the same source, the local SPAN session goes down when you save the configuration and reload the switch.
- The drop action is not supported with the VLAN access-map configuration with the filter access-group for a monitor session. The monitor session goes into an error state if the VLAN access-map with a drop action is configured with the filter access-group in the monitor session.
- Both permit and deny ACEs are treated alike. Packets that match the ACE are mirrored irrespective of whether they have a permit or deny entry in the ACL.
- ERSPAN is not supported for management ports.
- A destination port can be configured in only one ERSPAN session at a time.
- You cannot configure a port as both a source and destination port.
- A single ERSPAN session can include mixed sources in any combination of the following:
 - Ethernet ports or port channels but not subinterfaces.
 - VLANs or port channels, which can be assigned to port channel subinterfaces.
 - Port channels to the control plane CPU.



Note ERSPAN does not monitor any packets that are generated by the supervisor, regardless of their source.

- Destination ports do not participate in any spanning tree instance or Layer 3 protocols.
- When an ERSPAN session contains source ports that are monitored in the transmit or transmit and receive direction, packets that these ports receive may be replicated to the ERSPAN destination port although the packets are not actually transmitted on the source ports. Some examples of this behavior on source ports are as follows:
 - Traffic that results from flooding
 - Broadcast and multicast traffic

- For VLAN ERSPAN sessions with both ingress and egress that is configured, two packets (one from ingress and one from egress) are forwarded from the destination port if the packets get switched on the same VLAN.
- VLAN ERSPAN monitors only the traffic that leaves or enters Layer 2 ports in the VLAN.
- When the Cisco Nexus 3600 platform switch is the ERSPAN destination, GRE headers are not stripped off before sending mirrored packets out of the terminating point. Packets are sent along with the GRE headers as GRE packets and the original packet as the GRE payload.
- The egress interface for the ERSPAN source session is now printed in the output of the **show monitor session <session-number>** CLI command. The egress interface can be a physical port or a port-channel. For ECMP, one interface among the ECMP members is displayed in the output. This particular interface is used for the traffic egress.
- TCAM carving is not required for SPAN/ERSPAN on Cisco Nexus 3600 platform switches.
- You can view the SPAN/ERSPAN ACL statistics using the **show monitor filter-list** command. The output of the command displays all the entries along with the statistics from the SPAN TCAM. The ACL name is not printed, but only the entries are printed in the output. You can clear the statistics using the **clear monitor filter-list statistics** command. The output is similar to **show ip access-list** command. The Cisco Nexus 3600 platform switch does not provide support per ACL level statistics. This enhancement is supported for both local SPAN and ERSPAN.
- The traffic to and/or from the CPU is spanned. It is similar to any other interface SPAN. This enhancement is supported only in local SPAN. It is not supported with ACL source. The Cisco Nexus 3600 platform switch does not span the packets with (RCPU.dest_port != 0) header that is sent out from the CPU.
- For SPAN forward drop traffic, SPAN only the packets that get dropped due to various reasons in the forwarding plane. This enhancement is supported only for ERSPAN Source session. It is not supported along with SPAN ACL, Source VLAN, and Source interface. Three ACL entries are installed to SPAN dropped traffic. Priority can be set for the drop entries to have a higher or lower priority than the SPAN ACL entries and the VLAN SPAN entries of the other monitor sessions. By default, the drop entries have a higher priority.
- SPAN UDF (User-Defined Field) based ACL support
 - You can match any packet header or payload (certain length limitations) in the first 128 bytes of the packet.
 - You can define the UDFs with particular offset and length to match.
 - You can match the length as 1 or 2 bytes only.
 - Maximum of 8 UDFs are supported.
 - Additional UDF match criteria is added to ACL.
 - The UDF match criteria can be configured only for SPAN ACL. This enhancement is not supported for other ACL features, for example, RACL, PACL, and VACL.
 - Each ACE can have up to 8 UDF match criteria.
 - The UDF and http-redirect configuration should not coexist in the same ACL.
 - The UDF names need to be qualified for the SPAN TCAM.
 - The UDFs are effective only if they are qualified by the SPAN TCAM.

- The configuration for the UDF definition and the UDF name qualification in the SPAN TCAM require the use of **copy r s** command and reload.
 - The UDF match is supported for both Local SPAN and ERSPAN Src sessions.
 - The UDF name can have a maximum length of 16 characters.
 - The UDF offset starts from 0 (zero). If offset is specified as an odd number, 2 UDFs are used in the hardware for one UDF definition in the software. The configuration is rejected if the number of UDFs usage in the hardware goes beyond 8.
 - The UDF match requires the SPAN TCAM region to go double-wide. Therefore, you have to reduce the other TCAM regions' size to make space for SPAN.
 - The SPAN UDFs are not supported in tap-aggregation mode.
- If a sup-eth source interface is configured in the erspan-src session, the acl-span cannot be added as a source into that session and vice versa.
 - IPv6 User Defined Field (UDF) on ERSPAN support
 - ERSPAN source and ERSPAN destination sessions must use dedicated loopback interfaces. Such loopback interfaces should not be having any control plane protocols.

Default Settings for ERSPAN

The following table lists the default settings for ERSPAN parameters.

Table 23: 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. **no monitor session** {*session-number* | **all**}
4. **monitor session** {*session-number* | **all**} **type erspan-source**
5. **description** *description*
6. **filter access-group** *acl-name*
7. **source** {**interface** *type* [**rx** | **tx** | **both**] | **vlan** {*number* | *range*} [**rx**]}
8. (Optional) Repeat Step 6 to configure all ERSPAN sources.
9. (Optional) **filter access-group** *acl-filter*
10. **destination ip** *ip-address*
11. (Optional) **ip ttl** *ttl-number*
12. (Optional) **ip dscp** *dscp-number*
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

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# config t switch(config)#</pre>	Enters global configuration mode.
Step 2	monitor erspan origin ip-address <i>ip-address</i> global Example: <pre>switch(config)# monitor erspan origin ip-address 10.0.0.1 global</pre>	Configures the ERSPAN global origin IP address.
Step 3	no monitor session { <i>session-number</i> all } Example: <pre>switch(config)# no monitor session 3</pre>	Clears the configuration of the specified ERSPAN session. The new session configuration is added to the existing session configuration.
Step 4	monitor session { <i>session-number</i> all } type erspan-source Example: <pre>switch(config)# monitor session 3 type erspan-source switch(config-erspan-src)#</pre>	Configures an ERSPAN source session.

	Command or Action	Purpose
Step 5	description <i>description</i> Example: <pre>switch(config-erspan-src)# description erspan_src_session_3</pre>	Configures a description for the session. By default, no description is defined. The description can be up to 32 alphanumeric characters.
Step 6	filter access-group <i>acl-name</i> Example: <pre>switch(config-erspan-src)# filter access-group acl1</pre>	Filters ingress traffic at source ports based on the ACL list. Only packets that match the access list are spanned. The <i>acl-name</i> is an IP access-list, but not an access-map.
Step 7	source { interface type [rx tx both] vlan { <i>number</i> <i>range</i> } [rx]} Example: <pre>switch(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx</pre> Example: <pre>switch(config-erspan-src)# source interface port-channel 2</pre> Example: <pre>switch(config-erspan-src)# source interface sup-eth 0 both</pre> Example: <pre>switch(config-monitor)# source interface ethernet 101/1/1-3</pre>	
Step 8	(Optional) Repeat Step 6 to configure all ERSPAN sources.	—
Step 9	(Optional) filter access-group <i>acl-filter</i> Example: <pre>switch(config-erspan-src)# filter access-group ACL1</pre>	Associates an ACL with the ERSPAN session. Note You can create an ACL using the standard ACL configuration process. For more information, see the Cisco Nexus NX-OS Security Configuration Guide for your platform.
Step 10	destination ip <i>ip-address</i> Example: <pre>switch(config-erspan-src)# destination ip 10.1.1.1</pre>	Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session.
Step 11	(Optional) ip ttl <i>ttl-number</i> Example: <pre>switch(config-erspan-src)# ip ttl 25</pre>	Configures the IP time-to-live (TTL) value for the ERSPAN traffic. The range is from 1 to 255.
Step 12	(Optional) ip dscp <i>dscp-number</i> Example: <pre>switch(config-erspan-src)# ip dscp 42</pre>	Configures the differentiated services code point (DSCP) value of the packets in the ERSPAN traffic. The range is from 0 to 63.

	Command or Action	Purpose
Step 13	no shut Example: <pre>switch(config-erspan-src)# no shut</pre>	Enables the ERSPAN source session. By default, the session is created in the shut state. Note Only two ERSPAN source sessions can be running simultaneously.
Step 14	(Optional) show monitor session {all <i>session-number</i> range <i>session-range</i> } Example: <pre>switch(config-erspan-src)# show monitor session 3</pre>	Displays the ERSPAN session configuration.
Step 15	(Optional) show running-config monitor Example: <pre>switch(config-erspan-src)# show running-config monitor</pre>	Displays the running ERSPAN configuration.
Step 16	(Optional) show startup-config monitor Example: <pre>switch(config-erspan-src)# show startup-config monitor</pre>	Displays the ERSPAN startup configuration.
Step 17	(Optional) copy running-config startup-config Example: <pre>switch(config-erspan-src)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring SPAN Forward Drop Traffic for ERSPAN Source Session

SUMMARY STEPS

1. **configure terminal**
2. **monitor session** {*session-number* | all} **type erspan-source**
3. **vrf** *vrf-name*
4. **destination ip** *ip-address*
5. **source forward-drops rx** [*priority-low*]
6. **no shut**
7. (Optional) **show monitor session** {all | *session-number* | range *session-range*}

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	monitor session { <i>session-number</i> all } type erspan-source Example: <pre>switch(config)# monitor session 1 type erspan-source switch(config-erspan-src)#</pre>	Configures an ERSPAN source session.
Step 3	vrf <i>vrf-name</i> Example: <pre>switch(config-erspan-src)# vrf default</pre>	Configures the VRF that the ERSPAN source session uses for traffic forwarding.
Step 4	destination ip <i>ip-address</i> Example: <pre>switch(config-erspan-src)# destination ip 10.1.1.1</pre>	Configures the destination IP address in the ERSPAN session. Only one destination IP address is supported per ERSPAN source session.
Step 5	source forward-drops rx [<i>priority-low</i>] Example: <pre>switch(config-erspan-src)# source forward-drops rx [priority-low]</pre>	Configures the SPAN forward drop traffic for the ERSPAN source session. When configured as a low priority, this SPAN ACE matching drop condition takes less priority over any other SPAN ACEs configured by the interface ACL SPAN or VLAN ACL SPAN. Without the priority-low keyword, these drop ACEs take high priority compared to the regular interface or the VLAN SPAN ACLs. The priority matters only when the packet matching drop ACEs and the interface/VLAN SPAN ACLs are configured.
Step 6	no shut Example: <pre>switch(config-erspan-src)# no shut</pre>	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 7	(Optional) show monitor session { all <i>session-number</i> range session-range } Example: <pre>switch(config-erspan-src)# show monitor session 3</pre>	Displays the ERSPAN session configuration.

Example

```
switch# config t
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 40.1.1.1
switch(config-erspan-src)# source forward-drops rx
switch(config-erspan-src)# no shut
switch(config-erspan-src)# show monitor session 1

switch# config t
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 40.1.1.1
switch(config-erspan-src)# source forward-drops rx priority-low
```

```
switch(config-erspan-src) # no shut
switch(config-erspan-src) # show monitor session 1
```

Configuring an ERSPAN ACL

You can create an IPv4 ERSPAN ACL on the device and add rules to it.

Before you begin

To modify the DSCP value or the GRE protocol, you need to allocate a new destination monitor session. A maximum of four destination monitor sessions are supported.

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list *acl-name***
3. [*sequence-number*] {**permit** | **deny**} *protocol source destination* [**set-erspan-dscp** *dscp-value*] [**set-erspan-gre-proto** *protocol-value*]
4. (Optional) **show ip access-lists *name***
5. (Optional) **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**]
6. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	ip access-list <i>acl-name</i> Example: switch(config)# ip access-list erspan-acl switch(config-acl)#	Creates the ERSPAN ACL and enters IP ACL configuration mode. The <i>acl-name</i> argument can be up to 64 characters.
Step 3	[<i>sequence-number</i>] { permit deny } <i>protocol source destination</i> [set-erspan-dscp <i>dscp-value</i>] [set-erspan-gre-proto <i>protocol-value</i>] Example: switch(config-acl)# permit ip 192.168.2.0/24 any set-erspan-dscp 40 set-erspan-gre-proto 5555	Creates a rule in the ERSPAN ACL. You can create many rules. The <i>sequence-number</i> argument can be a whole number between 1 and 4294967295. The permit and deny commands support many ways of identifying traffic. The set-erspan-dscp option sets the DSCP value in the ERSPAN outer IP header. The range for the DSCP value is from 0 to 63. The DSCP value configured in the ERSPAN ACL overrides the value configured in the monitor session. If you do not include this option in the ERSPAN ACL, 0 or the DSCP value configured in the monitor session will be set.

	Command or Action	Purpose
		<p>The set-erspan-gre-proto option sets the protocol value in the ERSPAN GRE header. The range for the protocol value is from 0 to 65535. If you do not include this option in the ERSPAN ACL, the default value of 0x88be will be set as the protocol in the GRE header for ERSPAN-encapsulated packets.</p> <p>Each access control entry (ACE) with the set-erspan-gre-proto or set-erspan-dscp action consumes one destination monitor session. A maximum of three ACEs with one of these actions is supported per ERSPAN ACL. For example, you can configure one of the following:</p> <ul style="list-style-type: none"> • One ERSPAN session with an ACL having a maximum of three ACEs with the set-erspan-gre-proto or set-erspan-dscp action • One ERSPAN session with an ACL having two ACEs with the set-erspan-gre-proto or set-erspan-dscp action and one additional local or ERSPAN session • A maximum of two ERSPAN sessions with an ACL having one ACE with the set-erspan-gre-proto or set-erspan-dscp action
Step 4	(Optional) show ip access-lists <i>name</i> Example: <pre>switch(config-acl)# show ip access-lists erspan-acl</pre>	Displays the ERSPAN ACL configuration.
Step 5	(Optional) show monitor session { all <i>session-number</i> range <i>session-range</i> } [brief] Example: <pre>switch(config-acl)# show monitor session 1</pre>	Displays the ERSPAN session configuration.
Step 6	(Optional) copy running-config startup-config Example: <pre>switch(config-acl)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Configuring User Defined Field (UDF) Based ACL Support

You can configure User Defined Field (UDF) based ACL support on Cisco Nexus 3600 platform switches. See the following steps to configure ERSPAN based on UDF. See the Guidelines and Limitations for ERSPAN section for more information.

SUMMARY STEPS

1. switch# **configure terminal**

2. switch(config)# **udf** < udf -name> <packet start> <offset> <length>
3. switch(config)# **udf** < udf -name> header <Layer3/Layer4> <offset> <length>
4. switch(config)# **hardware profile tcam region span qualify udf** <name1>..... <name8>
5. switch(config)# **permit** <regular ACE match criteria> **udf** <name1> < val > <mask> <name8> < val > <mask>
6. switch(config)# **show monitor session** <session-number>

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# udf < udf -name> <packet start> <offset> <length> Example: (config)# udf udf1 packet-start 10 2 (config)# udf udf2 packet-start 50 2	Defines the UDF. Note You can define multiple UDFs but it is recommended to configure only the required UDFs. This configuration takes affect only after attaching the UDFs to a TCAM region and rebooting the box, as the UDFs are added to a region's qualifier set at TCAM carving time (boot up time).
Step 3	switch(config)# udf < udf -name> header <Layer3/Layer4> <offset> <length> Example: (config)# udf udf3 header outer 14 0 1 (config)# udf udf3 header outer 14 10 2 (config)# udf udf3 header outer 14 50 1	Defines the UDF.
Step 4	switch(config)# hardware profile tcam region span qualify udf <name1>..... <name8> Example: (config)# hardware profile tcam region span qualify udf udf1 udf2 udf3 udf4 udf5 [SUCCESS] Changes to UDF qualifier set will be applicable only after reboot. You need to 'copy run start' and 'reload' config)#	Configure UDF Qualification in SPAN TCAM. Add the UDFs to qualifier set for a TCAM region at TCAM carving time (happens at boot up time). The configuration allows maximum 4 UDFs that can be attached to a span region, all UDFs listed in a single command for a region. A new configuration for a region replaces the current configuration, but note that it needs a reboot for the configuration to come to the effect. When the UDF qualifier is added to the SPAN TCAM, the TCAM region expands from single wide to double wide. Make sure enough free space (128 more single wide entries) is available for the expansion or else the command gets rejected. Re-enter the command after creating the space by reducing TCAM space from the unused regions. Once the UDFs are detached from SPAN/TCAM region using the no hardware profile tcam region span qualify udf <name1> ..<name8> command, the SPAN TCAM region is considered as a single wide entry.
Step 5	switch(config)# permit <regular ACE match criteria> udf <name1> < val > <mask> <name8> < val > <mask>	Configure an ACL with UDF match.

	Command or Action	Purpose
	<p>Example:</p> <pre>(config)# ip access-list test 10 permit ip any any udf udf1 0x1234 0xffff udf3 0x56 0xff 30 permit ip any any dscp af11 udf udf5 0x22 0x22 config)#</pre>	
Step 6	<p>switch(config)# show monitor session <session-number></p> <p>Example:</p> <pre>(config)# show monitor session 1 session 1 ----- type : erspan-source state : up vrf-name : default destination-ip : 40.1.1.1 ip-ttl : 255 ip-dscp : 0 acl-name : test origin-ip : 100.1.1.10 (global) source intf : rx : Eth1/20 tx : Eth1/20 both : Eth1/20 source VLANs : rx : source fwd drops : egress-intf : Eth1/23 switch# config)#</pre>	<p>Displays the ACL using the show monitor session <session-number> command. You can check if the SPAN TCAM region is carved or not using the BCM SHELL command.</p>

Configuring IPv6 User Defined Field (UDF) on ERSPAN

You can configure IPv6 User Defined Field (UDF) on ERSPAN on Cisco Nexus 3600 platform switches. See the following steps to configure ERSPAN based on IPv6 UDF. See the Guidelines and Limitations for ERSPAN section for more information

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **udf** < udf -name> <packet start> <offset> <length>
3. switch(config)# **udf** < udf -name> header <Layer3/Layer4> <offset> <length>
4. switch(config)# **hardware profile tcam region ipv6-span-l2 512**
5. switch(config)# **hardware profile tcam region ipv6-span 512**
6. switch(config)# **hardware profile tcam region span spanv6 qualify udf** <name1>..... <name8>
7. switch(config)# **hardware profile tcam region span spanv6-12 qualify udf** <name1>..... <name8>
8. switch (config-erspan-src)# **filter** ipv6 access-group.... <aclname>.... <allow-sharing>
9. switch(config)# **permit** <regular ACE match criteria> **udf** <name1> < val > <mask> <name8> < val > <mask>
10. switch(config)# **show monitor session** <session-number>

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# udf < udf -name> <packet start> <offset> <length> Example: (config)# udf udf1 packet-start 10 2 (config)# udf udf2 packet-start 50 2	Defines the UDF. Note You can define multiple UDFs but it is recommended to configure only the required UDFs. This configuration takes affect only after attaching the UDFs to a TCAM region and rebooting the box, as the UDFs are added to a region's qualifier set at TCAM carving time (boot up time).
Step 3	switch(config)# udf < udf -name> header <Layer3/Layer4> <offset> <length> Example: (config)# udf udf3 header outer 14 0 1 (config)# udf udf3 header outer 14 10 2 (config)# udf udf3 header outer 14 50 1	Defines the UDF.
Step 4	switch(config)# hardware profile tcam region ipv6-span-l2 512 Example: (config)# hardware profile tcam region ipv6-span-l2 512 Warning: Please save config and reload the system for the configuration to take effect. config)#	Configure IPv6 on UDF on layer 2 ports. A new configuration for a region replaces the current configuration and you must reboot the switch for the configuration to come to the effect.
Step 5	switch(config)# hardware profile tcam region ipv6-span 512 Example: (config)# hardware profile tcam region ipv6-span 512 Warning: Please save config and reload the system for the configuration to take effect. config)#	Configure IPv6 on UDF on layer 3 ports. A new configuration for a region replaces the current configuration and you must reboot the switch for the configuration to come to the effect.
Step 6	switch(config)# hardware profile tcam region span spanv6 qualify udf <name1>..... <name8> Example: (config)# hardware profile tcam region spanv6 qualify udf udf1 [SUCCESS] Changes to UDF qualifier set will be applicable only after reboot. You need to 'copy run start' and	Configure UDF Qualification in SPAN for layer 3 ports. This enables the UDF match for ipv6-span TCAM region. Add the UDFs to qualifier set for a TCAM region at TCAM carving time (happens at boot up time). The configuration allows maximum of 2 IPv6 UDFs that can be attached to a SPAN region, all UDFs listed in a single command for a region. A new configuration for a region replaces the current configuration, but note that it needs a reboot for the configuration to come to the effect.

	Command or Action	Purpose
	'reload' config)#	
Step 7	switch(config)# hardware profile tcam region span spanv6-12 qualify udf <name1>..... <name8> Example: (config)# hardware profile tcam region spanv6-12 qualify udf udf1 [SUCCESS] Changes to UDF qualifier set will be applicable only after reboot. You need to 'copy run start' and 'reload' config)#	Configure UDF Qualification in SPAN for layer 2 ports. This enables the UDF match for ipv6-span-12 TCAM region. Add the UDFs to qualifier set for a TCAM region at TCAM carving time (happens at boot up time). The configuration allows a maximum of 2 IPv6 UDFs that can be attached to a SPAN region, all UDFs listed in a single command for a region. A new configuration for a region replaces the current configuration, but note that it needs a reboot for the configuration to come to the effect.
Step 8	switch (config-erspan-src)# filter <i>ipv6 access-group</i> <aclname>.... <allow-sharing> Example: (config-erspan-src)# ipv6 filter access-group test (config)#	Configure a IPv6 ACL in SPAN and ERSPAN mode. You can have only one of “filter ip access-group” or “filter ipv6 access-group” configuration in one monitor session. If same source interface is part of a IPv4 and IPv6 ERSPAN ACL monitor session, the “allow-sharing” needs to be configured with the “filter [ipv6] access-group” in the monitor session configuration.
Step 9	switch(config)# permit <regular ACE match criteria> udf <name1> < val > <mask> <name8> < val > <mask> Example: (config-erspan-src)# ipv6 access-list test (config-ipv6-acl)# permit ipv6 any any udf udf1 0x1 0x0	Configure an ACL with UDF match.
Step 10	switch(config)# show monitor session <session-number> Example: (config)# show monitor session 1 session 1 ----- type : erspan-source state : up vrf-name : default destination-ip : 40.1.1.1 ip-ttl : 255 ip-dscp : 0 acl-name : test origin-ip : 100.1.1.10 (global) source intf : rx : Eth1/20 tx : Eth1/20 both : Eth1/20 source VLANs : filter VLANs : filter not specified rx : source fwd drops : egress-intf : Eth1/23 switch# config)#	Displays the ACL using the show monitor session <session-number> command.

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

	Command or Action	Purpose
Step 1	configuration terminal Example: <pre>switch# configuration terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	monitor session { <i>session-range</i> all } shut Example: <pre>switch(config)# monitor session 3 shut</pre>	Shuts down the specified ERSPAN sessions. The session range is from 1-18. By default, sessions are created in the shut state. Four unidirectional sessions or two bidirectional sessions can be active at the same time. Note <ul style="list-style-type: none"> • In Cisco Nexus 5000 and 5500 platforms, two sessions can run simultaneously. • In Cisco Nexus 5600 and 6000 platforms, 16 sessions can run simultaneously.
Step 3	no monitor session { <i>session-range</i> all } shut Example:	Resumes (enables) the specified ERSPAN sessions. The session range is from 1-18. The session range is from 1-18. By default, sessions are created in the shut state. Four

	Command or Action	Purpose
	<code>switch(config)# no monitor session 3 shut</code>	<p>unidirectional sessions or two bidirectional sessions can be active at the same time.</p> <p>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.</p>
Step 4	<p>monitor session <i>session-number</i> type erspan-source</p> <p>Example:</p> <pre>switch(config)# monitor session 3 type erspan-source switch(config-erspan-src)#</pre>	Enters the monitor configuration mode for the ERSPAN source type. The new session configuration is added to the existing session configuration.
Step 5	<p>monitor session <i>session-number</i> type erspan-destination</p> <p>Example:</p> <pre>switch(config-erspan-src)# monitor session 3 type erspan-destination</pre>	Enters the monitor configuration mode for the ERSPAN destination type.
Step 6	<p>shut</p> <p>Example:</p> <pre>switch(config-erspan-src)# shut</pre>	Shuts down the ERSPAN session. By default, the session is created in the shut state.
Step 7	<p>no shut</p> <p>Example:</p> <pre>switch(config-erspan-src)# no shut</pre>	Enables the ERSPAN session. By default, the session is created in the shut state.
Step 8	<p>(Optional) show monitor session all</p> <p>Example:</p> <pre>switch(config-erspan-src)# show monitor session all</pre>	Displays the status of ERSPAN sessions.
Step 9	<p>(Optional) show running-config monitor</p> <p>Example:</p> <pre>switch(config-erspan-src)# show running-config monitor</pre>	Displays the running ERSPAN configuration.
Step 10	<p>(Optional) show startup-config monitor</p> <p>Example:</p> <pre>switch(config-erspan-src)# show startup-config monitor</pre>	Displays the ERSPAN startup configuration.
Step 11	<p>(Optional) copy running-config startup-config</p> <p>Example:</p> <pre>switch(config-erspan-src)# copy running-config startup-config</pre>	Copies the running configuration to the startup configuration.

Verifying the ERSPAN Configuration

Use the following command to verify the ERSPAN configuration information:

Command	Purpose
show monitor session { all <i>session-number</i> range <i>session-range</i> }	Displays the ERSPAN session configuration.
show running-config monitor	Displays the running ERSPAN configuration.
show startup-config monitor	Displays the ERSPAN startup configuration.

Configuration Examples for ERSPAN

Configuration Example for an ERSPAN Source Session

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

```
switch# config t
switch(config)# interface e14/30
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# monitor erspan origin ip-address 3.3.3.3 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# filter access-group acl1
switch(config-erspan-src)# source interface e14/30
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 9.1.1.2
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
switch(config)# show monitor session 1
```

Configuration Example for an ERSPAN ACL

This example shows how to configure an ERSPAN ACL:

```
switch# configure terminal
switch(config)# ip access-list match_11_pkts
switch(config-acl)# permit ip 11.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# ip access-list match_12_pkts
switch(config-acl)# permit ip 12.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# vlan access-map erspan_filter 5
switch(config-access-map)# match ip address match_11_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# vlan access-map erspan_filter 10
switch(config-access-map)# match ip address match_12_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
```



```
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# filter access_group erspan_filter
```

Configuration Examples for UDF-Based ERSPAN

This example shows how to configure UDF-based ERSPAN to match on the inner TCP flags of an encapsulated IP-in-IP packet using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + Outer IP (20) + Inner IP (20) + Inner TCP (20, but TCP flags at 13th byte)
- Offset from packet-start: 14 + 20 + 20 + 13 = 67
- UDF match value: 0x20
- UDF mask: 0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify udf udf_tcpflags
copy running-config startup-config
reload
ip access-list acl-udf
 permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1 type erspan-source
 source interface Ethernet 1/1
 filter access-group acl-udf
```

This example shows how to configure UDF-based ERSPAN to match regular IP packets with a packet signature (DEADBEEF) at 6 bytes after a Layer 4 header start using the following match criteria:

- Outer source IP address: 10.0.0.2
- Inner TCP flags: Urgent TCP flag is set
- Bytes: Eth Hdr (14) + IP (20) + TCP (20) + Payload: 112233445566DEADBEEF7788
- Offset from Layer 4 header start: 20 + 6 = 26
- UDF match value: 0xDEADBEEF (split into two-byte chunks and two UDFs)
- UDF mask: 0xFFFFFFFF

```
udf udf_pktsig_msb header outer 13 26 2
udf udf_pktsig_lsb header outer 13 28 2
hardware access-list tcam region racl qualify udf udf_pktsig_msb udf_pktsig_lsb
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
 permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1 type erspan-source
 source interface Ethernet 1/1
 filter access-group acl-udf-pktsig
```

Additional References

Related Documents

Related Topic	Document Title
ERSPAN commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	<i>Cisco Nexus NX-OS System Management Command Reference</i> for your platform.



CHAPTER 18

Configuring DNS

This chapter contains the following sections:

- [About DNS Client](#) , on page 217
- [Prerequisites for DNS Clients](#), on page 218
- [Default Settings for DNS Clients](#), on page 218
- [Configuring the DNS Source Interface](#), on page 218
- [Configuring DNS Clients](#), on page 219

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 Nexus 3600 platform switches supports stateless restarts for the DNS client. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

Prerequisites for DNS Clients

The DNS client has the following prerequisites:

- You must have a DNS name server on your network.

Default Settings for DNS Clients

The following table shows the default settings for DNS client parameters.

Parameter	Default
DNS client	Enabled

Configuring the DNS Source Interface

You can configure DNS to use a specific interface.

SUMMARY STEPS

1. `switch# configure terminal`
2. `switch(config)# ip dns source-interface type slot/port`
3. `switch(config)# show ip dns source-interface`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>switch# configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# ip dns source-interface <i>type slot/port</i>	<p>Configures the source interface for all DNS packets. The following list contains the valid values for <i>interface</i>.</p> <ul style="list-style-type: none"> • ethernet • loopback • mgmt • port-channel • vlan <p>Note When you, configure the source interface for DNS, SCP copy operations initiated from the server fail. To perform an SCP copy operation from the server, remove the DNS source interface configuration.</p>
Step 3	switch(config)# show ip dns source-interface	Displays the configured DNS source interface.

Example

This example shows how to configure the DNS source interface:

```
switch(config)# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ip dns source-interface ethernet 1/8
switch(config)# show ip dns source-interface
VRF Name                Interface
default                  Ethernet1/8
```

Configuring DNS Clients

You can configure the DNS client to use a DNS server on your network.

Before you begin

- Ensure that you have a domain name server on your network.

SUMMARY STEPS

1. switch# **configuration terminal**
2. switch(config)# **vrf context management**
3. switch(config)# **{ip | ipv6} host name ip/ipv6 address1 [ip/ipv6 address2... ip/ipv6 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 ip/ipv6 server-address1 [ip/ipv6 server-address2... ip/ipv6 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

	Command or Action	Purpose
Step 1	switch# configuration terminal	Enters global configuration mode.
Step 2	switch(config)# vrf context managment	Specifies a configurable virtual and routing (VRF) name.
Step 3	switch(config)# { ip ipv6 } host name <i>ip/ipv6 address1</i> [<i>ip/ipv6 address2... ip/ipv6 address6</i>]	Defines up to six static hostname-to-address mappings in the host name cache.
Step 4	(Optional) switch(config)# ip domain name <i>name</i> [use-vrf <i>vrf-name</i>]	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 <i>name</i> [use-vrf <i>vrf-name</i>]	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 <i>ip/ipv6 server-address1</i> [<i>ip/ipv6 server-address2... ip/ipv6 server-address6</i>] [use-vrf <i>vrf-name</i>]	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.
Step 9	switch(config)# exit	Exits configuration mode and returns to EXEC mode.
Step 10	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following example shows how to configure a default domain name and enable DNS lookup:

```
switch# config t
switch(config)# vrf context management
switch(config)# ip domain-name mycompany.com
switch(config)# ip name-server 172.68.0.10
switch(config)# ip domain-lookup
```




CHAPTER 19

Configuring sFlow

This chapter contains the following sections:

- [About sFlow, on page 223](#)
- [Prerequisites, on page 224](#)
- [Guidelines and Limitations for sFlow, on page 224](#)
- [Default Settings for sFlow, on page 224](#)
- [Minimum Requirements for Sampling, on page 224](#)
- [Configuring sFlow, on page 225](#)
- [Verifying the sFlow Configuration, on page 232](#)
- [Configuration Examples for sFlow, on page 233](#)
- [Additional References for sFlow, on page 233](#)

About sFlow

sFlow allows you to monitor the real-time traffic in data networks that contain switches and routers. It uses the sampling mechanism in the sFlow Agent software on switches and routers for monitoring traffic and to forward the sample data on ingress and egress ports to the central data collector, also called the sFlow Analyzer.

For more information about sFlow, see RFC 3176.

sFlow Agent

The sFlow Agent, which is embedded in the Cisco NX-OS software, periodically samples or polls the interface counters that are associated with a data source of the sampled packets. The data source can be an Ethernet interface, an EtherChannel interface, or a range of either. Ethernet or port-channel sub-interfaces are not supported. The sFlow Agent queries the Ethernet port manager for the respective EtherChannel membership information and also receives notifications from the Ethernet port manager for membership changes.

When you enable sFlow sampling in the Cisco NX-OS software, based on the sampling rate and the hardware internal random number, the ingress packets and egress packets are sent to the CPU as an sFlow-sampled packet. The sFlow Agent processes the sampled packets and sends an sFlow datagram to the sFlow Analyzer. In addition to the original sampled packet, an sFlow datagram includes the information about the ingress port, egress port, and the original packet length. An sFlow datagram can have multiple sFlow samples.

Prerequisites

You must enable the sFlow feature using the **feature sflow** command to configure sFlow.

Guidelines and Limitations for sFlow

The sFlow configuration guidelines and limitations are as follows:

- When you enable sFlow for an interface, it is enabled for both ingress and egress. You cannot enable sFlow for only ingress or only egress.
- sFlow egress sampling for multicast, broadcast, or unknown unicast packets is not supported.
- You should configure the sampling rate based on the sFlow configuration and traffic in the system.
- Cisco Nexus 3600 platform switches supports only one sFlow collector.
- Ethernet or port-channel sub-interfaces are not supported as sFlow data-source ports.
- You cannot configure individual port-channel member ports as sFlow data-sources. The port-channel bundle interface can be sFlow enabled data-source ports, such as sFlow data-source interface po1.
- For Cisco Nexus N3K-C36180YC-R, N3K-C3636C-R, N9K-X9636C-RX, and N9K-X96136YC-R platform switches, egress sampled traffic will always have the first data-source interface in the list as Source ID index in sflow record.

Default Settings for sFlow

Table 24: Default sFlow Parameters

Parameters	Default
sFlow sampling-rate	4096
sFlow sampling-size	128
sFlow max datagram-size	1400
sFlow collector-port	6343
sFlow counter-poll-interval	20

Minimum Requirements for Sampling

Without these configured, no packets will be sampled: After you enable the sFlow feature, you must explicitly configure the following configuration elements for the packet sampling to take effect on the device.

- Sflow Agent-IP
- Sflow Collector-IP

- Sflow Data-source interface

If you do not configure the configuration elements, packets will not be sampled.

The default configuration elements specified as the default settings for sFlow are optional.

Configuring sFlow

Enabling the sFlow Feature

You must enable the sFlow feature before you can configure sFlow on the switch.

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] feature sflow	Enables the sFlow feature.
Step 3	(Optional) show feature	Displays enabled and disabled features.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to enable the sFlow feature:

```
switch# configure terminal
switch(config)# feature sflow
switch(config)# copy running-config startup-config
```

Configuring the Sampling Rate

Before you begin

Ensure that you have enabled the sFlow feature.

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **sflow sampling-rate** *sampling-rate*
3. (Optional) **show sflow**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] sflow sampling-rate <i>sampling-rate</i>	Configures the sFlow sampling rate for packets. The <i>sampling-rate</i> can be an integer between 4096-1000000000. The default value is 4096.
Step 3	(Optional) show sflow	Displays sFlow information.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to set the sampling rate to 50,000:

```
switch# configure terminal
switch(config)# sflow sampling-rate 50000
switch(config)# copy running-config startup-config
```

With the above configuration, approximately 1 out of every 50,000 packets will be sampled and sent to the sFlow collector. Note that there could be a slight variance.

Configuring the Maximum Sampled Size

You can configure the maximum number of bytes that should be copied from a sampled packet.

Before you begin

Ensure that you have enabled the sFlow feature.

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **sflow max-sampled-size** *sampling-size*
3. (Optional) **show sflow**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] sflow max-sampled-size <i>sampling-size</i>	Configures the sFlow maximum sampling size packets. The range for the <i>sampling-size</i> is from 64 to 256 bytes. The default value is 128.
Step 3	(Optional) show sflow	Displays configured sFlow values.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the maximum sampling size for the sFlow Agent:

```
switch# configure terminal
switch(config)# sflow max-sampled-size 200
switch(config)# copy running-config startup-config
```

Configuring the Counter Poll Interval

You can configure the maximum number of seconds between successive samples of the counters that are associated with the data source. A sampling interval of 0 disables counter sampling.

Before you begin

Ensure that you have enabled the sFlow feature.

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **sflow counter-poll-interval** *poll-interval*
3. (Optional) **show sflow**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] sflow counter-poll-interval <i>poll-interval</i>	Configures the sFlow poll interval for an interface. The range for the <i>poll-interval</i> is from 0 to 2147483647 seconds. The default value is 20. Configuring 0 disables the counter polling.
Step 3	(Optional) show sflow	Displays sFlow information.

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

Example

This example shows how to configure the sFlow poll interval for an interface:

```
switch# configure terminal
switch(config)# sflow counter-poll-interval 100
switch(config)# copy running-config startup-config
```

Configuring the Maximum Datagram Size

You can configure the maximum number of data bytes that can be sent in a single sample datagram.

Before you begin

Ensure that you have enabled the sFlow feature.

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **sflow max-datagram-size** *datagram-size*
3. (Optional) **show sflow**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] sflow max-datagram-size <i>datagram-size</i>	Configures the sFlow maximum datagram size. The range for the <i>datagram-size</i> is from 200 to 9000 bytes. The default value is 1400.
Step 3	(Optional) show sflow	Displays configured sFlow values.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the sFlow maximum datagram size:

```

switch# configure terminal
switch(config)# sflow max-datagram-size 2000
switch(config)# copy running-config startup-config
[#####] 100%

```

Configuring the sFlow Analyzer Address

Before you begin

Ensure that you have enabled the sFlow feature.

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **sflow collector-ip vrf IP-address vrf-instance**
3. (Optional) **show sflow**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] sflow collector-ip vrf IP-address vrf-instance	Configures the IPv4 address for the sFlow Analyzer. <i>vrf-instance</i> can be one of the following: <ul style="list-style-type: none"> • A user-defined VRF name—You can specify a maximum of 32 alphanumeric characters. • vrf management— You must use this option if the sFlow data collector is on the network connected to the management port. • vrf default—You must use this option if the sFlow data collector is connected to a network reachable via any front panel port residing on the default vrf.
Step 3	(Optional) show sflow	Purpose "Displays configured sFlow values.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the IPv4 address of the sFlow data collector that is connected to the management port:

```
switch# configure terminal
switch(config)# sflow collector-ip 192.0.2.5 vrf management
switch(config)# copy running-config startup-config
```

Configuring the sFlow Analyzer Port

You can configure the destination port for sFlow datagrams.

Before you begin

Ensure that you have enabled the sFlow feature.

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **sflow collector-port** *collector-port*
3. (Optional) **show sflow**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] sflow collector-port <i>collector-port</i>	Configures the UDP port of the sFlow Analyzer. The range for the <i>collector-port</i> is from 0 to 65535. The default value is 6343.
Step 3	(Optional) show sflow	Displays configured sFlow values.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the destination port for sFlow datagrams:

```
switch# configure terminal
switch(config)# sflow collector-port 7000
switch(config)# copy running-config startup-config
[#####] 100%
switch(config)#
```

Configuring the sFlow Agent Address

Before you begin

Ensure that you have enabled the sFlow feature.

SUMMARY STEPS

1. switch# **configure terminal**
2. [no] **sflow agent-ip** *ip-address*
3. (Optional) **show sflow**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	[no] sflow agent-ip <i>ip-address</i>	Configures the IPv4 address of the sFlow Agent. The default <i>ip-address</i> is 0.0.0.0, which means that all sampling is disabled on the switch. You must specify a valid IP address to enable sFlow functionality. The configured value can be an IP address present on the local system or any other arbitrary IP value desired for a tracking purpose.
Step 3	(Optional) show sflow	Displays sFlow information.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the IPv4 address of the sFlow Agent:

```
switch# configure terminal
switch(config)# sflow agent-ip 192.0.2.3
switch(config)# copy running-config startup-config
```

Configuring the sFlow Sampling Data Source

The sFlow sampling data source can be an Ethernet port, a range of Ethernet ports, or a port channel.

Before you begin

- Ensure that you have enabled the sFlow feature.
- If you want to use a port channel as the data source, ensure that you have already configured the port channel and you know the port channel number.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# [no] **sflow data-source interface** [*ethernet slot/port[-port]*] | **port-channel** *channel-number*]
3. (Optional) switch(config)# **show sflow**
4. (Optional) switch(config)# **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config)# [no] sflow data-source interface [ethernet slot/port[-port] port-channel channel-number]	Configures the sFlow sampling data source. For an Ethernet data source, <i>slot</i> is the slot number and <i>port</i> can be either a single port number or a range of ports designated as <i>port-port</i> .
Step 3	(Optional) switch(config)# show sflow	Displays configured sFlow values.
Step 4	(Optional) switch(config)# copy running-config startup-config	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure Ethernet ports 5 through 12 for the sFlow sampler:

```
switch# configure terminal
switch(config)# sflow data-source interface ethernet 1/5-12
switch(config)# copy running-config startup-config
[#####] 100%
switch(config)#
```

This example shows how to configure port channel 100 for the sFlow sampler:

```
switch# configure terminal
switch(config)# sflow data-source interface port-channel 100
switch(config)# copy running-config startup-config
[#####] 100%
switch(config)#
```

Verifying the sFlow Configuration

Use the following commands to verify the sFlow configuration information:

Command	Purpose
show sflow	Displays the sFlow global configuration.
show sflow statistics	Displays the sFlow statistics.
clear sflow statistics	Clears the sFlow statistics.
show running-config sflow [all]	Displays the current running sFlow configuration.

Configuration Examples for sFlow

This example shows how to configure sFlow:

```
feature sflow
sflow sampling-rate 5000
sflow max-sampled-size 200
sflow counter-poll-interval 100
sflow max-datagram-size 2000
sflow collector-ip 192.0.2.5 vrf management
sflow collector-port 7000
sflow agent-ip 192.0.2.3
sflow data-source interface ethernet 1/5
```

Additional References for sFlow

Table 25: Related Documents for sFlow

Related Topic	Document Title
sFlow CLI commands	<i>Cisco Nexus 3600 NX-OS Command Reference.</i>
RFC 3176	Defines the sFlow packet format and SNMP MIB. http://www.sflow.org/rfc3176.txt



CHAPTER 20

Configuring Graceful Insertion and Removal

This chapter contains the following sections:

- [About Graceful Insertion and Removal, on page 235](#)
- [GIR Workflow, on page 237](#)
- [Configuring the Maintenance-Mode Profile, on page 238](#)
- [Configuring the Normal-Mode Profile, on page 239](#)
- [Creating a Snapshot, on page 240](#)
- [Adding Show Commands to Snapshots, on page 242](#)
- [Triggering Graceful Removal, on page 243](#)
- [Triggering Graceful Insertion, on page 246](#)
- [Maintenance Mode Enhancements, on page 247](#)
- [Verifying the GIR Configuration, on page 248](#)

About Graceful Insertion and Removal

You can use graceful insertion and removal to gracefully eject a switch and isolate it from the network in order to perform debugging or upgrade operations. The switch is removed from the regular forwarding path with minimal traffic disruption. When you are finished performing debugging or upgrade operations, you can use graceful insertion to return the switch to its fully operational (normal) mode.

In graceful removal, all protocols and vPC domains are gracefully brought down and the switch is isolated from the network. In graceful insertion, all protocols and vPC domains are restored.

The following protocols are supported (for both IPv4 and IPv6 address families):

- Border Gateway Protocol (BGP)
- Enhanced Interior Gateway Routing Protocol (EIGRP)
- Intermediate System-to-Intermediate System (ISIS)
- Open Shortest Path First (OSPF)
- Protocol Independent Multicast (PIM)
- Routing Information Protocol (RIP)



Note For graceful insertion and removal, the PIM protocol is applicable only to vPC environments. During graceful removal, the vPC forwarding role is transferred to the vPC peer for all northbound sources of multicast traffic.

Profiles

By default, the system isolates all enabled protocols during graceful removal and restores them during graceful insertion. The protocols are isolated and restored in a predefined order.

If you want to isolate, shut down, or restore the protocols individually (or perform additional configurations), you can create a profile with configuration commands that can be applied during graceful removal or graceful insertion. However, you need to make sure that the order of the protocols is correct and any dependencies are considered.

The switch supports the following profiles:

- Maintenance-mode profile—Contains all the commands that will be executed during graceful removal, when the switch enters maintenance mode.
- Normal-mode profile—Contains all the commands that will be executed during graceful insertion, when the switch returns to normal mode.

The following commands (along with any configuration commands) are supported in the profiles:

Command	Description
isolate	Isolates the protocol from the switch and puts the protocol in maintenance mode.
no isolate	Restores the protocol and puts the protocol in normal mode.
shutdown	Shuts down the protocol or vPC domain.
no shutdown	Brings up the protocol or vPC domain.
system interface shutdown [exclude fex-fabric]	Shuts down the system interfaces (except the management interface).
no system interface shutdown [exclude fex-fabric]	Brings up the system interfaces.
sleep instance <i>instance-number seconds</i>	<p>Delays the execution of the command by a specified number of seconds. You can delay multiple instances of the command.</p> <p>The range for the <i>instance-number</i> and <i>seconds</i> arguments is from 0 to 2177483647.</p>

Command	Description
python instance <i>instance-number uri</i> [<i>python-arguments</i>] Example: python instance 1 bootflash://script1.py	Configures Python script invocations to the profile. You can add multiple invocations of the command to the profile. You can enter a maximum of 32 alphanumeric characters for the Python arguments.

Snapshots

In Cisco NX-OS, a snapshot is the process of capturing the running states of selected features and storing them on persistent storage media.

Snapshots are useful to compare the state of a switch before graceful removal and after graceful insertion. The snapshot process consists of three parts:

- Creating a snapshot of the states of a few preselected features on the switch and storing them on the persistent storage media
- Listing the snapshots taken at various time intervals and managing them
- Comparing snapshots and showing the differences between features

GIR Workflow

Follow these steps to complete the graceful insertion and removal (GIR) workflow:

1. (Optional) Create the maintenance-mode profile. (See [Configuring the Maintenance-Mode Profile, on page 238.](#))
2. (Optional) Create the normal-mode profile. (See [Configuring the Normal-Mode Profile, on page 239.](#))
3. Take a snapshot before triggering graceful removal. (See [Creating a Snapshot, on page 240.](#))
4. Trigger graceful removal to put the switch in maintenance mode. (See [Triggering Graceful Removal, on page 243.](#))
5. Trigger graceful insertion to return the switch to normal mode. (See [Triggering Graceful Insertion, on page 246.](#))
6. Take a snapshot after triggering graceful insertion. (See [Creating a Snapshot, on page 240.](#))
7. Use the **show snapshots compare** command to compare the operational data before and after the graceful removal and insertion of the switch to make sure that everything is running as expected. (See [Verifying the GIR Configuration, on page 248.](#))

Configuring the Maintenance-Mode Profile

You can create a maintenance-mode profile with configuration commands that can be applied during graceful removal or graceful insertion.

SUMMARY STEPS

1. **configure maintenance profile maintenance-mode**
2. **end**
3. **show maintenance profile maintenance-mode**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure maintenance profile maintenance-mode Example: <pre>switch# configure maintenance profile maintenance-mode Enter configuration commands, one per line. End with CNTL/Z. switch(config-mm-profile)#</pre>	Enters a configuration session for the maintenance-mode profile. Depending on which protocols you have configured, you must now enter the appropriate commands to bring down the protocols. For a list of supported commands, see Profiles, on page 236 .
Step 2	end Example: <pre>switch(config-mm-profile)# end switch#</pre>	Closes the maintenance-mode profile.
Step 3	show maintenance profile maintenance-mode Example: <pre>switch# show maintenance profile maintenance-mode</pre>	Displays the details of the maintenance-mode profile.

Example

This example shows how to create a maintenance-mode profile:

```
switch# configure maintenance profile maintenance-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# ip pim isolate
switch(config-mm-profile)# vpc domain 10
switch(config-mm-profile-config-vpc-domain)# shutdown
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# shutdown
switch(config-mm-profile)# router eigrp 10
switch(config-mm-profile-router)# shutdown
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# shutdown
switch(config-mm-profile)# system interface shutdown
switch(config-mm-profile)# end
Exit maintenance profile mode.
switch# show maintenance profile maintenance-mode
[Maintenance Mode]
```



```

ip pim isolate
vpc domain 10
  shutdown
router bgp 100
  shutdown
router eigrp 10
  shutdown
  address-family ipv6 unicast
    shutdown
system interface shutdown

```

Configuring the Normal-Mode Profile

You can create a normal-mode profile with configuration commands that can be applied during graceful removal or graceful insertion.

SUMMARY STEPS

1. **configure maintenance profile normal-mode**
2. **end**
3. **show maintenance profile normal-mode**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure maintenance profile normal-mode Example: <pre>switch# configure maintenance profile normal-mode Enter configuration commands, one per line. End with CNTL/Z. switch(config-mm-profile)#</pre>	Enters a configuration session for the normal-mode profile. Depending on which protocols you have configured, you must now enter the appropriate commands to bring up the protocols. For a list of supported commands, see Profiles, on page 236 .
Step 2	end Example: <pre>switch(config-mm-profile)# end switch#</pre>	Closes the normal-mode profile.
Step 3	show maintenance profile normal-mode Example: <pre>switch# show maintenance profile normal-mode</pre>	Displays the details of the normal-mode profile.

Example

This example shows how to create a maintenance-mode profile:

```

switch# configure maintenance profile normal-mode
switch(config-mm-profile)# no system interface shutdown
switch(config-mm-profile)# router eigrp 10
switch(config-mm-profile-router)# no shutdown

```

```

switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# no shutdown
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# no shutdown
switch(config-mm-profile)# vpc domain 10
switch(config-mm-profile-config-vpc-domain)# no shutdown
switch(config-mm-profile)# no ip pim isolate
switch(config-mm-profile)# end
Exit maintenance profile mode.
switch# show maintenance profile normal-mode
[Normal Mode]
no system interface shutdown
router eigrp 10
  no shutdown
  address-family ipv6 unicast
  no shutdown
router bgp 100
  no shutdown
vpc domain 10
  no shutdown
no ip pim isolate

```

Creating a Snapshot

You can create a snapshot of the running states of selected features. When you create a snapshot, a predefined set of **show** commands are run and the outputs are saved.

SUMMARY STEPS

1. **snapshot create** *snapshot-name description*
2. **show snapshots**
3. **show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>snapshot create <i>snapshot-name description</i></p> <p>Example:</p> <pre> switch# snapshot create snap_before_maintenance Taken before maintenance Executing 'show interface'... Done Executing 'show ip route summary vrf all'... Done Executing 'show ipv6 route summary vrf all'... Done Executing 'show bgp sessions vrf all'... Done Executing 'show ip eigrp topology summary'... Done Executing 'show ipv6 eigrp topology summary'... Done Feature 'vpc' not enabled, skipping... Executing 'show ip ospf vrf all'... Done Feature 'ospfv3' not enabled, skipping... Feature 'isis' not enabled, skipping... Feature 'rip' not enabled, skipping... Snapshot 'snap_before_maintenance' created </pre>	<p>Captures the running state or operational data of selected features and stores the data on persistent storage media.</p> <p>You can enter a maximum of 64 alphanumeric characters for the snapshot name and a maximum of 254 alphanumeric characters for the description.</p> <p>Use the snapshot delete {all <i>snapshot-name</i>} command to delete all snapshots or a specific snapshot.</p>

	Command or Action	Purpose
Step 2	<p>show snapshots</p> <p>Example:</p> <pre>switch# show snapshots Snapshot Name Time Description -----</pre> <pre>snap_before_maintenance Wed Aug 19 13:53:28 2015 Taken before maintenance</pre>	Displays snapshots present on the switch.
Step 3	<p>show snapshots compare snapshot-name-1 snapshot-name-2 [summary ipv4routes ipv6routes]</p> <p>Example:</p> <pre>switch# show snapshots compare snap_before_maintenance snap_after_maintenance</pre>	<p>Displays a comparison of two snapshots.</p> <p>The summary option displays just enough information to see the overall changes between the two snapshots.</p> <p>The ipv4routes and ipv6routes options display the changes in IPv4 and IPv6 routes between the two snapshots.</p>

Example

The following example shows a summary of the changes between two snapshots:

```
switch# show snapshots compare snapshot1 snapshot2 summary
feature                               snapshot1  snapshot2  changed
basic summary
  # of interfaces                      16        12         *
  # of vlans                           10        4          *
  # of ipv4 routes                      33        3          *
.....

interfaces
  # of eth interfaces                  3         0          *
  # of eth interfaces up                2         0          *
  # of eth interfaces down              1         0          *
  # of eth interfaces other             0         0

  # of vlan interfaces                  3         1          *
  # of vlan interfaces up                3         1          *
  # of vlan interfaces down              0         0
  # of vlan interfaces other            0         1          *
.....
```

The following example shows the changes in IPv4 routes between two snapshots:

```
switch# show snapshots compare snapshot1 snapshot2 ipv4routes
metric                               snapshot1  snapshot2  changed
# of routes                          33        3          *
# of adjacencies                      10        4          *

Prefix                               Changed Attribute
-----
23.0.0.0/8                           not in snapshot2
10.10.10.1/32                         not in snapshot2
21.1.2.3/8                            adjacency index has changed from 29 (snapshot1) to 38 (snapshot2)
.....
```

There were 28 attribute changes detected

Adding Show Commands to Snapshots

You can specify additional **show** commands to be captured in snapshots. These **show** commands are defined in user-specified snapshot sections.

SUMMARY STEPS

1. **snapshot section add** *section "show-command" row-id element-key1 [element-key2]*
2. **show snapshots sections**
3. **show snapshots compare** *snapshot-name-1 snapshot-name-2 [summary | ipv4routes | ipv6routes]*

DETAILED STEPS

	Command or Action	Purpose
Step 1	snapshot section add <i>section "show-command" row-id element-key1 [element-key2]</i> Example: <pre>switch# snapshot section add myshow "show ip interface brief" ROW_intf intf-name</pre>	<p>Adds a user-specified section to snapshots. The <i>section</i> is used to name the show command output. You can use any word to name the section.</p> <p>The show command must be enclosed in quotation marks. Non-show commands will not be accepted.</p> <p>The <i>row-id</i> argument specifies the tag of each row entry of the show command's XML output. The <i>element-key1</i> and <i>element-key2</i> arguments specify the tags used to distinguish among row entries. In most cases, only the <i>element-key1</i> argument needs to be specified to be able to distinguish among row entries.</p> <p>Note To delete a user-specified section from snapshots, use the snapshot section delete <i>section</i> command.</p>
Step 2	show snapshots sections Example: <pre>switch# show snapshots sections</pre>	Displays the user-specified snapshot sections.
Step 3	show snapshots compare <i>snapshot-name-1 snapshot-name-2 [summary ipv4routes ipv6routes]</i> Example: <pre>switch# show snapshots compare snap1 snap2</pre>	<p>Displays a comparison of two snapshots.</p> <p>The summary option displays just enough information to see the overall changes between the two snapshots.</p> <p>The ipv4routes and ipv6routes options display the changes in IPv4 and IPv6 routes between the two snapshots.</p>

Example

The following example adds the **show ip interface brief** command to the myshow snapshot section. It also compares two snapshots (snap1 and snap2) and shows the user-specified sections in both snapshots.

```

switch# snapshot section add myshow "show ip interface brief" ROW_intf intf-name
switch# show snapshots sections
user-specified snapshot sections
-----
[myshow]
  cmd: show ip interface brief
  row: ROW_intf
  key1: intf-name
  key2: -

[sect2]
  cmd: show ip ospf vrf all
  row: ROW_ctx
  key1: instance_number
  key2: cname

switch# show snapshots compare snap1 snap2
=====
Feature                Tag                snap1                snap2
=====

[bgp]
-----

[interface]
-----

      [interface:mgmt0]
            vdc_lvl_in_pkts          692310                **692317**
            vdc_lvl_in_mcast        575281                **575287**
            vdc_lvl_in_bcast        77209                 **77210**
            vdc_lvl_in_bytes        63293252              **63293714**
            vdc_lvl_out_pkts        41197                 **41198**
            vdc_lvl_out_ucast       33966                 **33967**
            vdc_lvl_out_bytes       6419714               **6419788**
-----

[ospf]
-----

[myshow]
-----

      [interface:Ethernet1/1]
            state                    up                    **down**
            admin_state              up                    **down**
-----

```

Triggering Graceful Removal

In order to perform debugging or upgrade operations, you can trigger a graceful removal of the switch, which will eject the switch and isolate it from the network.

Before you begin

If you want the system to use a maintenance-mode profile that you create, see [Configuring the Maintenance-Mode Profile, on page 238](#).

SUMMARY STEPS

1. **configure terminal**
2. **system mode maintenance** [**dont-generate-profile** | **timeout** *value* | **shutdown** | **on-reload reset-reason** *reason*]
3. (Optional) **show system mode**
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	system mode maintenance [dont-generate-profile timeout <i>value</i> shutdown on-reload reset-reason <i>reason</i>] Example: <pre>switch(config)# system mode maintenance Following configuration will be applied: ip pim isolate router bgp 65502 isolate router ospf p1 isolate router ospfv3 p1 isolate Do you want to continue (y/n)? [no] y Generating a snapshot before going into maintenance mode Starting to apply commands... Applying : ip pim isolate Applying : router bgp 65502 Applying : isolate Applying : router ospf p1 Applying : isolate Applying : router ospfv3 p1 Applying : isolate Maintenance mode operation successful.</pre>	Puts all enabled protocols in maintenance mode (using the isolate command). The following options are available: <ul style="list-style-type: none"> • dont-generate-profile—Prevents the dynamic searching of enabled protocols and executes commands configured in a maintenance-mode profile. Use this option if you want the system to use a maintenance-mode profile that you have created. • timeout <i>value</i>—Keeps the switch in maintenance mode for a specified number of minutes. The range is from 5 to 65535. Once the configured time elapses, the switch returns to normal mode automatically. The no system mode maintenance timeout command disables the timer. • shutdown—Shuts down all protocols, vPC domains, and interfaces except the management interface (using the shutdown command). This option is disruptive while the default (which uses the isolate command) is not. • on-reload reset-reason <i>reason</i>—Boots the switch into maintenance mode automatically in the event of a specified system crash. The no system mode maintenance on-reload reset-reason command prevents the switch from being brought up in maintenance mode in the event of a system crash. The maintenance mode reset reasons are as follows:

	Command or Action	Purpose
		<ul style="list-style-type: none"> • HW_ERROR—Hardware error • SVC_FAILURE—Critical service failure • KERN_FAILURE—Kernel panic • WDOG_TIMEOUT—Watchdog timeout • FATAL_ERROR—Fatal error • LC_FAILURE—Line card failure • MATCH_ANY—Any of the above reasons <p>The system prompts you to continue. Enter y to continue or n to terminate the process.</p>
Step 3	<p>(Optional) show system mode</p> <p>Example:</p> <pre>switch(config)# show system mode System Mode: Maintenance</pre>	<p>Displays the current system mode.</p> <p>The switch is in maintenance mode. You can now perform any desired debugging or upgrade operations on the switch.</p>
Step 4	<p>(Optional) copy running-config startup-config</p> <p>Example:</p> <pre>switch(config)# copy running-config startup-config</pre>	<p>Copies the running configuration to the startup configuration. This command is required if you want to preserve maintenance mode following a reboot.</p>

Example

This example shows how to shut down all protocols, vPC domains, and interfaces on the switch:

```
switch(config)# system mode maintenance shutdown
```

Following configuration will be applied:

```
vpc domain 10
 shutdown
router bgp 65502
 shutdown
router ospf p1
 shutdown
router ospfv3 p1
 shutdown
system interface shutdown
```

Do you want to continue (y/n)? [no] **y**

Generating a snapshot before going into maintenance mode

Starting to apply commands...

```
Applying : vpc domain 10
Applying : shutdown
Applying : router bgp 65502
Applying : shutdown
Applying : router ospf p1
```

```
Applying : shutdown
Applying : router ospfv3 p1
Applying : shutdown

Maintenance mode operation successful.
```

This example shows how to automatically boot the switch into maintenance mode if a fatal error occurs:

```
switch(config)# system mode maintenance on-reload reset-reason fatal_error
```

Triggering Graceful Insertion

When you finish performing any debugging or upgrade operations, you can trigger a graceful insertion to restore all protocols.

Before you begin

If you want the system to use a normal-mode profile that you create, see [Configuring the Maintenance-Mode Profile, on page 238](#).

SUMMARY STEPS

1. **configure terminal**
2. **no system mode maintenance [dont-generate-profile]**
3. (Optional) **show system mode**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	no system mode maintenance [dont-generate-profile] Example: <pre>switch(config)# no system mode maintenance dont-generate-profile Following configuration will be applied: no ip pim isolate router bgp 65502 no isolate router ospf p1 no isolate router ospfv3 p1 no isolate Do you want to continue (y/n)? [no] y</pre>	<p>Puts all enabled protocols in normal mode (using the no isolate command).</p> <p>The dont-generate-profile option prevents the dynamic searching of enabled protocols and executes commands configured in a normal-mode profile. Use this option if you want the system to use a normal-mode profile that you have created.</p> <p>The system prompts you to continue. Enter y to continue or n to terminate the process.</p>

	Command or Action	Purpose
	<pre>Starting to apply commands... Applying : no ip pim isolate Applying : router bgp 65502 Applying : no isolate Applying : router ospf p1 Applying : no isolate Applying : router ospfv3 p1 Applying : no isolate Maintenance mode operation successful. Generating Current Snapshot</pre>	
Step 3	<p>(Optional) show system mode</p> <p>Example:</p> <pre>switch(config)# show system mode System Mode: Normal</pre>	Displays the current system mode. The switch is now in normal mode and is fully operational.

Maintenance Mode Enhancements

The following maintenance mode enhancements are added to Cisco Nexus 3600 platform switches:

- In the system maintenance shutdown mode, the following message is added:

```
NOTE: The command system interface shutdown will shutdown all interfaces excluding mgmt 0.
```

- Entering the CLI command, **system mode maintenance** checks and sends alerts for the orphan ports.
- In isolate mode, when the vPC is configured, the following message is added:

```
NOTE: If you have vPC orphan interfaces, please ensure vpc orphan-port suspend is configured under them, before proceeding further.
```

- Custom Profile Configuration: A new CLI command, **system mode maintenance always-use-custom-profile** is added for custom profile configuration. A new CLI command, **system mode maintenance non-interactive** is added under `#ifdef` for Cisco Nexus 9000 Series switches only.

When you create a custom profile (in maintenance or normal mode), it displays the following message:

```
Please use the command system mode maintenance always-use-custom-profile if you want to always use the custom profile.
```

- A delay has been added before the `after_maintenance` snapshot is taken. The **no system mode maintenance** command exits once all the configuration for the normal mode has been applied, the mode has been changed to normal mode, and a timer has been started to take the `after_maintenance` snapshot. Once the timer expires, the `after_maintenance` snapshot is taken in the background and a new warning syslog, `MODE_SNAPSHOT_DONE` is sent once the snapshot is complete.

The final output of the CLI command **no system mode maintenance** indicates when the `after_maintenance` snapshot is generated:

```
The after_maintenance snapshot will be generated in <delay> seconds.
After that time, please use show snapshots compare before_maintenance
after_maintenance to check the health of the system. The timer delay
```

for the `after_maintenance` snapshot is defaulted to 120 seconds but it can be changed by a new configuration command.

The new configuration command to change the timer delay for the `after_maintenance` snapshot is **system mode maintenance snapshot-delay <seconds>**. This configuration overrides the default setting of 120 seconds to any value between 0 and 65535 and it is displayed in the ASCII configuration.

A new show command, **show maintenance snapshot-delay** has also been added to display the current snapshot-delay value. This new show command supports the XML output.

- A visible CLI indicator has been added to display when the system is in the maintenance mode, for example, `switch (m-mode) #`.
- Support for the SNMP traps has been added when the device moves from the maintenance mode to the normal mode and vice-versa through CLI reload, or system reset. The **snmp-server enable traps mmode cseMaintModeChangeNotify** trap is added to enable changing to the maintenance mode trap notification. The **snmp-server enable traps mmode cseNormalModeChangeNotify** is added to enable changing to the normal mode trap notification. Both the traps are disabled by default.

Verifying the GIR Configuration

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

Command	Purpose
show interface brief	Displays abbreviated interface information.
show maintenance on-reload reset-reasons	Displays the reset reasons for which the switch comes up in maintenance mode. For a description of the maintenance mode reset reasons, see Triggering Graceful Removal, on page 243 .
show maintenance profile [maintenance-mode normal-mode]	Displays the details of the maintenance-mode or normal-mode profile.
show maintenance timeout	Displays the maintenance-mode timeout period, after which the switch automatically returns to normal mode.
show { running-config startup-config } mmode [all]	Displays the maintenance-mode section of the running or startup configuration. The all option includes the default values.
show snapshots	Displays snapshots present on the switch.
show snapshots compare <i>snapshot-name-1</i> <i>snapshot-name-2</i> [summary ipv4routes ipv6routes]	<p>Displays a comparison of two snapshots.</p> <p>The summary option displays just enough information to see the overall changes between the two snapshots.</p> <p>The ipv4routes and ipv6routes options display the changes in IPv4 and IPv6 routes between the two snapshots.</p>

Command	Purpose
show snapshots dump <i>snapshot-name</i>	Displays the content of each file that was generated when the snapshot was taken.
show snapshots sections	Displays the user-specified snapshot sections.
show system mode	Displays the current system mode.



CHAPTER 21

Performing Configuration Replace

This chapter includes the following sections:

- [About Configuration Replace and Commit-timeout, on page 251](#)
- [Overview, on page 251](#)
- [Guidelines and Limitations for Configuration Replace, on page 253](#)
- [Recommended Workflow for Configuration Replace, on page 255](#)
- [Performing a Configuration Replace, on page 256](#)
- [Verifying Configuration Replace, on page 258](#)
- [Examples for Configuration Replace, on page 258](#)

About Configuration Replace and Commit-timeout

The configuration replace feature enables you to replace the running configuration of the Cisco Nexus switch with the user provided configuration without reloading the device. The device reload may be required only when a configuration itself requires a reload. The running configuration file that is provided by the user should be taken using copy running file. Unlike **copy file: to running**, the configuration replace feature is not a merge operation. This feature replaces the entire running configuration with a new configuration that is provided by the user. If there is a failure in the configuration replace, the original configuration is restored in the switch. From Cisco NX-OS Release 9.3(1), **best-effort** option is introduced. This option enables the configuration replace to execute the full patch despite any error in the commands and the original configuration is not restored in the switch.

The commit-timeout feature enables you to rollback to the previous configuration after successfully performing the configuration replace operation. If the commit timer expires, the rollback operation is automatically initiated.



Note

- You must provide a valid running configuration that has been received with the Cisco NX-OS device. It should not be a partial configuration.
-

Overview

The configuration replace feature has the following operation steps:

- Configuration replace intelligently calculates the difference between the current running-configuration and the user-provided configuration in the Cisco Nexus switch and generates a patch file which is the difference between the two files. You can view this patch file which includes a set of configuration commands.
- Configuration replace applies the configuration commands from the patch file similarly to executing commands.
- The configuration rolls back to or restores the previous running configuration under the following situations:
 - If there is a mismatch in the configuration after the patch file has been applied.
 - If you perform the configuration operation with a commit timeout and the commit timer expires.
- The configuration does not roll back to or does not restore the previous running configuration when the best-effort option is used. This option enables the configuration replace to execute the full patch despite any error in the commands and will not roll back to the previous configuration.
- You can view the exact configuration that caused a failure using the **show config-replace log exec** command.
- Restore operations that fail while restoring the switch to the original configuration, are not interrupted. The restore operation continues with the remaining configuration. Use the **show config-replace log exec** command to list the commands that failed during the restore operation.
- If you enter the **configure replace commit** command before the timer expires, the commit timer stops and the switch runs on the user provided configuration that has been applied through the configuration replace feature.
- If the commit timer expires, roll back to the previous configuration is initiated automatically.
- In Cisco NX-OS Release 9.3(1), semantic validation support is added for the configuration replace. This semantic validation is done as part of the precheck in configuration replace. The patch gets applied only when the semantic validation is successful. After applying the patch file, configuration replace triggers the verification process. The configuration replace compares the running-configuration with the user configuration file during the verification process. If there is a mismatch, it restores the device to the original configuration.

The differences between configuration replace and copying a file to the running-configuration are as follows:

Configuration Replace	Copying a file
The configure replace <i><target-url></i> command removes the commands from the current running-configuration that are not present in the replacement file. It also adds commands that need to be added to the current running-configuration.	The copy <i><source-url></i> running-config command is a merge operation which preserves all the commands from, both the source file and the current running-configuration. This command does not remove the commands from the current running-configuration that are not present in the source file.
You must use a complete Cisco NX-OS configuration file as the replacement file for the configure replace <i><target-url></i> command.	You can use a partial configuration file as a source file for the copy <i><source-url></i> running-config command.

Benefits of Configuration Replace

The benefits of configuration replace are:

- You can replace the current running-configuration file with the user-provided configuration file without having to reload the switch or manually undo CLI changes to the running-configuration file. As a result, the system downtime is reduced.
- You can revert to the saved Cisco NX-OS configuration state.
- It simplifies the configuration changes by allowing you to apply a complete configuration file to the device, where only the commands that need to be added or removed are affected. The other service and configurations that are not modified remain untouched.
- If you configure the commit-timeout feature, you can rollback to the previous configuration even when the configuration replace operation has been successful.

Guidelines and Limitations for Configuration Replace

The configuration replace feature has the following configuration guidelines and limitations:

- The configuration replace feature is supported on Cisco Nexus 3000 Series and Cisco Nexus 9000 Series switches.
- Only one user can perform the configuration replace, checkpoint, and rollback operations, or copy the running-configuration to the startup configuration at the same time. Parallel operations such as operations via multiple Telnet, SSH, or NX-API sessions are not supported. The multiple configuration replace or rollback request is serialized, for example, only after the first request is completed, processing of the second request begins.
- You are not allowed to initiate another configuration replace operation when the commit timer is running. You must either stop the timer by using the **configure replace commit** command or wait until the commit timer expires before you initiate another configuration replace operation.
- 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.
- The user provided configuration file must be the valid show running-configuration output that is taken from the Cisco NX-OS device (copy run file). The configuration cannot be a partial configuration and must include mandated commands, such as user admin and so on.
- We do not recommend a configuration replace operation that is performed on the configuration file that is generated across the software version because this operation could fail. A new configuration file must be regenerated whenever there is change in the software version.
- We recommend that you do not change any configuration from others sessions if the configuration replace operation is in progress because it could cause the operation to fail.
- Note the following about the configuration replace feature:
 - The configuration replace feature does not support features that require a reload. One such feature is: system vlan reserve.

- The configuration replace feature is not supported on Cisco Nexus 9500 platform switches with -R line cards.
- The configuration replace feature could fail if the running configuration includes the **feature-set mpls** or the **mpls static range** commands and tries to move to a configuration without MPLS or modifies the label range.
- The configuration replace feature does not support autoconfigurations.
- If the line card to which the configuration replace feature is applied is offline, the configuration replace operation fails.
- An ITD service must be shut down (**shutdown**) prior to making ITD changes with the configuration replace feature.
- Sequence number is mandatory for CLI **ip community-list** and **ip as-path access-list** commands. Without a sequence number, the configuration replace operation fails.
- If your configurations demand reloading the Cisco NX-OS device in order to apply the configuration, then you must reload these configurations after the configuration replace operation.
- The order of the commands in the user provided configuration file must be the same as those commands in the running configuration of the Cisco Nexus switch.
- The user configuration file to which you need to replace the running configuration on the switch using CR should be generated from the running-config of the switch after configuring the new commands. The user configuration file should not be manually edited with the CLI commands and the sequence of the configuration commands should not be altered.
- The semantic validation is not supported in 4-Gig memory platforms.
- When different versions of a feature are present in the running configuration and user configuration (for example: VRRPv2 and VRRPv3), semantic validation option does not work as expected. This issue is a known limitation.
- Beginning from Cisco 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.
- 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).

Guidelines and Limitations for 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
<p>Use Case 1: Multiple command variants - Only the last command variant is retained</p> <p>The candidate config is automatically converted as shown in the third column before the CR patch is generated.</p>	<pre>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</pre>	<pre>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</pre>
<p>Use Case 2: Commands comprising track IDs - Only the last command variant with same next-hop and different track ID is retained</p> <p>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.</p>	<pre>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</pre>	<pre>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</pre>

Recommended Workflow for Configuration Replace

The following workflow is the recommended workflow for configuration replace:



Note

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

1. Generate a configuration file by first applying the configurations on a Cisco Nexus Series device and then use the **show running-configuration** output as the configuration file. Use this file to make configuration modifications as required. Then use this generated or updated configuration file to perform configuration replace.
2. View and verify the patch file by executing the **configure replace <file> show-patch** command. This is an optional step.

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

Performing a Configuration Replace

To perform configuration replace, do the following:

SUMMARY STEPS

1. **configure replace** { *<uri_local>* | *<uri_remote>* } [**verbose** | **show-patch**]
2. **configure replace** [**bootflash / scp / sftp**] *<user-configuration-file>* **show-patch**
3. **configure replace** [**bootflash / scp / sftp**] *<user-configuration-file>* **verbose**
4. **configure replace** *<user-configuration-file>* [**best-effort**]
5. **configure replace** *<user-configuration-file>* [**verify-and-commit**]
6. **configure replace** *<user-configuration-file>* [**verify-only**]
7. (Optional) **configure replace** [**bootflash / scp / sftp**] *<user-configuration-file>* **verbose** **commit-timeout** *<time>*
8. (Optional) **configure replace** [**commit**]
9. (Optional) **configure replace** [**bootflash/scp/sftp**] *<user-configuration-file>* *non-interactive*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>configure replace { <uri_local> <uri_remote> } [verbose show-patch]</code>	Performs configuration replace. If you make the configuration changes through any sessions when configuration replace is in progress, the configuration replace operation fails. If you send a configuration replace request when one configuration request is already in progress, then it gets serialized.
Step 2	<code>configure replace [bootflash / scp / sftp] <user-configuration-file> show-patch</code>	Displays the differences between the running-configuration and the user-provided configuration. Note <ul style="list-style-type: none"> • 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	<code>configure replace [bootflash / scp / sftp] <user-configuration-file> verbose</code>	Replaces the configuration on the switch with the new user configuration that is provided by the user. Configuration replace is always atomic.
Step 4	<code>configure replace <user-configuration-file> [best-effort]</code>	Replaces the configuration on the switch with the new user configuration and enables the configuration replace with semantic validation. The best-effort option enables the configuration replace to execute the full patch despite any error in the commands and also make sure that the previous configuration is not rolled back.
Step 5	<code>configure replace <user-configuration-file> [verify-and-commit]</code>	Replaces the configuration on the switch with the new user configuration and enables the configuration replace with semantic validation. The verify-and-commit option is used for enabling the semantic validation. Patch will be executed only if semantic validation of the full patch gets passed. You can use the best-effort option or the verify-and-commit option or both the options at the same time.
Step 6	<code>configure replace <user-configuration-file> [verify-only]</code>	Shows only the patch and does Semantic validation on the patch, and display the results. The patch does not get applied to the system.
Step 7	(Optional) <code>configure replace [bootflash / scp / sftp] <user-configuration-file> verbose commit-timeout <time></code>	Configures the commit time in seconds. The timer starts after the configuration replace operation is successfully completed.

	Command or Action	Purpose
Step 8	(Optional) configure replace [commit]	Stops the commit timer and continues the configuration replace configuration. Note This step is applicable only if you have configured the commit-timeout feature. Note To rollback to the previous configuration, you must wait for the expiry of the commit timer. Once the timer expires, the switch is automatically rolled back to the previous configuration.
Step 9	(Optional) configure replace [bootflash/scp/sftp] <i><user-configuration-file> non-interactive</i>	There is no user prompt in maintenance mode. The yes user-confirmation is taken by default, and rollback proceeds. You can use the non-interactive option only in the maintenance mode.

Verifying Configuration Replace

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

Table 26: Verifying Configuration Replace

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

Examples for Configuration Replace

See the following configuration examples for configuration replace:

- Use the **configure replace bootflash: <file> show-patch** CLI command to display the difference between the running-configurations and user-provided configurations.

```
switch(config)# configure replace bootflash:<file> show-patch
Collecting Running-Config
Converting to checkpoint file
#Generating Rollback Patch
!!
no role name abc
```

- Use the **configure replace bootflash: <file> verbose** CLI command to replace the entire running-configuration in the switch with the user-configuration.

```
switch(config)# configure replace bootflash:<file> verbose
Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallely may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
=====
config t
no role name abc
=====
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.
```

Sample Example with adding of BGP configurations.

```
switch(config)# sh run | section bgp
switch(config)# sh file bootflash:file | section bgp
feature bgp
router bgp 1
  address-family ipv4 unicast
  neighbor 1.1.1.1
switch(config)#
switch(config)# configure replace bootflash:file verbose
Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallely may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
=====
config t
feature bgp
router bgp 1
address-family ipv4 unicast
neighbor 1.1.1.1
=====
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.
```

```
switch(config)# sh run | section bgp
feature bgp
router bgp 1
  address-family ipv4 unicast
  neighbor 1.1.1.1
```

```

Sample Example with ACL
switch(config)# configure replace bootflash:run_1.txt
Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallelly may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
=====
config t
no ip access-list nexus-50-new-xyz
ip access-list nexus-50-new-xyz-jkl-abc
10 remark Newark
20 permit ip 17.31.5.0/28 any
30 permit ip 17.34.146.193/32 any
40 permit ip 17.128.199.0/27 any
50 permit ip 17.150.128.0/22 any
=====
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

switch(config)#

switch(config)# show run aclmgr | sec nexus-50-new-xyz-jkl-abc
ip access-list nexus-50-new-xyz-jkl-abc
 10 remark Newark
 20 permit ip 17.31.5.0/28 any
 30 permit ip 17.34.146.193/32 any
 40 permit ip 17.128.199.0/27 any
 50 permit ip 17.150.128.0/22 any

```

- Use the **configure replace bootflash:user-config.cfg verify-only** CLI command to generate and verify the patch semantically.

```

switch(config)# configure replace bootflash:user-config.cfg verify-only

Version match between user file and running configuration.
Pre-check for User config PASSED
Collecting Running-Config
Converting to checkpoint file
Generating Rollback Patch
Validating Patch
=====
`config t `
`interface Ethernet1/1`
`shutdown`
`no switchport trunk allowed vlan`
`no switchport mode`
`no switchport`
`exit`
Skip non dme command for CR validation
`interface Vlan1`
`shutdown`
`interface Ethernet1/1`
`shutdown`
`no switchport`
`ip address 1.1.1.1/24`
`exit`
Skip non dme command for CR validation
=====

```

```
Patch validation completed successful
switch(config)#
```

- Use the **configure replace bootflash:user-config.cfg best-effort verify-and-commit** CLI command to replace the switch running configuration with the given user configuration after performing the semantic validation on patch.

```
switch(config)# configure replace bootflash:user-config.cfg best-effort verify-and-commit
```

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

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

```
Validating Patch
Patch validation completed successful
Executing Rollback Patch
During CR operation, will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Rollback Patch
```

```
Configure replace completed successfully. Please run 'show config-replace log exec' to
see if there is any configuration that requires reload to take effect.
```

```
switch(config)#
```

- Use the **show config-replace log exec** CLI command to check all the configuration that is executed and failures if any.

```
switch(config)# show config-replace log exec
Operation          : Rollback to Checkpoint File
Checkpoint file name : .replace_tmp_28081
Scheme             : tmp
Rollback done By    : admin
Rollback mode       : atomic
Verbose             : enabled
Start Time          : Wed, 06:39:34 25 Jan 2017
```

```
-----
time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS
End Time           : Wed, 06:39:47 25 Jan 2017
Rollback Status    : Success
```

```
Executing Patch:
-----
switch#config t
switch#no role name abc
```

- Use the **show config-replace log verify** CLI command to check the failed configuration if any.

```
switch(config)# show config-replace log verify
Operation          : Rollback to Checkpoint File
Checkpoint file name : .replace_tmp_28081
Scheme             : tmp
```

```

Rollback done By      : admin
Rollback mode        : atomic
Verbose              : enabled
Start Time           : Wed, 06:39:34 25 Jan 2017
End Time             : Wed, 06:39:47 25 Jan 2017
Status               : Success

```

```

Verification patch contains the following commands:
-----

```

```

!!
! No changes
-----

```

```

time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS

```

- Use the **show config-replace status** CLI command to check the status of configuration replace.

```

switch(config)# show config-replace status
Last operation : Rollback to file
Details:
  Rollback type: atomic replace_tmp_28081
  Start Time: Wed Jan 25 06:39:28 2017
  End Time: Wed Jan 25 06:39:47 2017
  Operation Status: Success
switch(config)#

```

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

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

```

switch# show run all

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

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

```

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

```

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

version 9.3(1) Bios:version 05.39
hostname n9k13

```

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

```

switch# show file bootflash:test

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

```



```
!Time: Tue Nov 12 11:04:57 2019

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

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

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

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

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

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

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

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

```
Verification patch contains the following commands:
-----
```

```
!!
Configuration To Be Added Missing in Running-config
=====
!
power redundancy-mode ps-redundant

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

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



CHAPTER 22

Configuring Rollback

This chapter contains the following sections:

- [About Rollbacks, on page 265](#)
- [Guidelines and Limitations for Rollbacks, on page 265](#)
- [Creating a Checkpoint, on page 266](#)
- [Implementing a Rollback, on page 267](#)
- [Verifying the Rollback Configuration, on page 268](#)

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

	Command or Action	Purpose
Step 1	switch# checkpoint { [<i>cp-name</i>] [description <i>descr</i>] file <i>file-name</i>] 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- <i><number></i> where number is from 1 to 10. The description can contain up to 80 alphanumeric characters, including spaces.
Step 2	(Optional) switch# no checkpoint <i>cp-name</i> 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.
Step 3	(Optional) switch# show checkpoint <i>cp-name</i> Example:	Displays the contents of the checkpoint name.

	Command or Action	Purpose
	[all] switch# show checkpoint stable	

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

	Command or Action	Purpose
Step 1	show diff rollback-patch { checkpoint <i>src-cp-name</i> running-config startup-config file <i>source-file</i> } { checkpoint <i>dest-cp-name</i> running-config startup-config file <i>dest-file</i> } Example: 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 <i>cp-name</i> file <i>cp-file</i> } 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 <i>name</i> [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 <i>src-cp-name</i> running-config startup-config file <i>source-file</i> } { checkpoint <i>dest-cp-name</i> running-config startup-config file <i>dest-file</i> }	Displays the differences between the source and destination checkpoint selections.
show rollback log [exec verify]	Displays the contents of the rollback log.



Note Use the **clear checkpoint database** command to delete all checkpoint files.



CHAPTER 23

Integrity Check of Candidate Config

This chapter describes how to perform integrity check of Candidate Config.

This chapter includes the following sections:

- [About Candidate Config, on page 269](#)
- [Guidelines and Limitations for Candidate Config Integrity Check, on page 269](#)
- [Performing Integrity Check for Candidate Config, on page 274](#)
- [Examples of Integrity Check, on page 274](#)

About Candidate Config

Candidate config is a subset of the running-config which checks whether the Candidate config exists in the running-config without any additions or modifications or deletions.

To check the integrity of the candidate config, use the following commands:

- `show diff running-config`
- `show diff startup-config`

For more information on the CLIs, refer to [Performing Integrity Check for Candidate Config, on page 274](#).

Guidelines and Limitations for Candidate Config Integrity Check

Candidate config integrity check has the following guidelines and limitations:

- Beginning with Cisco NX-OS Release 10.2(3)F, Candidate config integrity check option is introduced on all Cisco Nexus switches.
- If you need to perform an integrity check on full running configuration as input instead of a partial config, then it is recommended not to use "**partial**" keyword.
- The line numbers that are displayed in the generated running config do not match with the candidate config as they are internally generated one.
- If there is any difference between the configuration of running and candidate, then it is displayed inline as output.

- If the whole block of configuration in the candidate file is a new addition, it will be appended at the end of the generated running config.
- Beginning from Cisco NX-OS Release 10.4(3)F, you can also use polymorphic commands in candidate configuration to perform partial diff.
- EIGRP address family IPv4 configs are recommended to configure under EIGRP address family hierarchy and not under router mode hierarchy in the candidate file, before executing 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 sub-modes, 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 non-case 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.
- Clear text passwords are allowed in the 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.

Guidelines and Limitations for Partial Diff of Default Commands for Multicast Components

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	<pre>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</pre>
PIM6	<pre>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</pre>
IGMP	<pre>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</pre>
MLD	<pre>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</pre>

Guidelines and Limitations for show diff running-config *file_url* [unified] [partial] [merged] command

- When using the **unified**, **partial**, and **merged** option to review the differences for the following PBR commands, the diff output 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**

- If the candidate next-hops are a subset of running next-hops (in same order and sequence), and candidate additive flags are a subset of running flags, then the diff output will be 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>

- If the candidate next-hops are a subset of running next-hops (in same order and sequence), and candidate has some extra additive flags which are not present in running config, then the diff output will append 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

- If candidate next-hops are not a subset of running next-hops (in same order and sequence), and the candidate and running record can have any additive flag, then the diff output will indicate 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 desired flags are preserved in the final, merged configuration.

Candidate Config	Running Config	Partial Unified Merged Diff Output
<pre>route-map rmap1 permit 10 set ip next-hop 1.1.1.1 2.2.2.2 load-share drop-on-fail</pre>	<pre>route-map rmap1 permit 10 set ip next-hop 2.2.2.2 1.1.1.1 load-share force-order</pre>	<pre>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</pre>

- 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
```

Prior to the generation of the partial-diff output, the above candidate configuration will be 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 **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.

Consequently, 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, in order 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 track ID is different for the same next-hop under candidate and running config, then the diff output will be as shown in the following table:

Candidate Config	Running Config	Partial Unified Merged Diff Output
<pre>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</pre>	<pre>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</pre>	<pre>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</pre>

2. If track ID is not present under candidate config but present in running config for the same next-hop, then the diff output will be empty as shown in the following table:

Candidate Config	Running Config	Partial Unified Merged Diff Output
<pre>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</pre>	<pre>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</pre>	<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 will be as shown in the following table:

Candidate Config	Running Config	Partial Unified Merged Diff Output
<pre>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</pre>	<pre>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</pre>	<pre>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</pre>

Performing Integrity Check for Candidate Config

To perform the integrity check, use the following commands:

Before you begin



Note Before performing the integrity check, ensure that the running config and the candidate config belong to the same image version.

SUMMARY STEPS

1. `show diff running-config file_url [unified] [partial] [merged]`
2. `show diff startup-config file_url [unified]`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>show diff running-config file_url [unified] [partial] [merged]</code></p> <p>Example:</p> <pre>switch# show diff running-config bootflash:candidate.cfg partial unified</pre>	<p>Displays the differences between the running and user given candidate config.</p> <ul style="list-style-type: none"> • <i>file_url</i>: File path to compare with. • unified: Displays the difference between running and user configuration in unified format. • partial: Enter partial only if user configuration file is partial and not a full configuration. • merged: Enter merged only if sub-commands need to be merged instead of replace.
Step 2	<p><code>show diff startup-config file_url [unified]</code></p> <p>Example:</p> <pre>switch# show diff startup-config bootflash:candidate.cfg unified</pre>	<p>Displays the differences between the startup and user given candidate config.</p> <ul style="list-style-type: none"> • <i>file_url</i>: File path to compare with. • unified: Displays the difference between startup and user configuration in unified format.

Examples of Integrity Check

No Difference Between Running and Candidate Config

```
switch# show diff running-config bootflash:base_running.cfg
switch#
```

Difference Between Running and Candidate

```

switch# show diff running-config bootflash:modified-running.cfg unified
--- running-config
+++ User-config
@@ -32,11 +32,11 @@

interface Ethernet1/1
    mtu 9100
    link debounce time 0
    beacon
-   ip address 2.2.2.2/24
+   ip address 1.1.1.1/24
    no shutdown

interface Ethernet1/2

interface Ethernet1/3
switch#

```

Difference Between Running and Partial Candidate

```

switch# show file bootflash:intf_vlan.cfg
interface Vlan101
    no shutdown
    no ip redirects
    ip address 1.1.2.1/24 secondary
    ip address 1.1.1.1/24
switch#
switch# show diff running-config bootflash:intf_vlan.cfg partial unified
--- running-config
+++ User-config
@@ -3897,10 +3883,14 @@
    mtu 9100
    ip access-group IPV4_EDGE in
    ip address 2.2.2.12/26 tag 54321

    interface Vlan101
+   no shutdown
+   no ip redirects
+   ip address 1.1.2.1/24 secondary
+   ip address 1.1.1.1/24

    interface Vlan102
        description Vlan102
        no shutdown
        mtu 9100
switch#

```

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#
```



CHAPTER 24

Configuring User Accounts and RBAC

This chapter contains the following sections:

- [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](#)
- [Default Settings for the User Accounts and RBAC, on page 286](#)

About User Accounts and RBAC

Cisco Nexus 3600 platform switches use role-based access control (RBAC) to define the amount of access that each user has when the user logs into the switch.

With RBAC, you define one or more user roles and then specify which management operations each user role is allowed to perform. When you create a user account for the switch, you associate that account with a user role, which then determines what the individual user is allowed to do on the switch.

User Roles

User roles contain rules that define the operations allowed for the user who is assigned the role. Each user role can contain multiple rules and each user can have multiple roles. For example, if role1 allows access only to configuration operations, and role2 allows access only to debug operations, users who belong to both role1 and role2 can access configuration and debug operations. You can also limit access to specific VLANs, and interfaces.

The switch provides the following default user roles:

network-admin (superuser)

Complete read and write access to the entire switch.

network-operator

Complete read access to the switch.



Note If you belong to multiple roles, you can execute a combination of all the commands permitted by these roles. Access to a command takes priority over being denied access to a command. For example, suppose a user has RoleA, which denied access to the configuration commands. However, the user also has RoleB, which has access to the configuration commands. In this case, the user has access to the configuration commands.

Rules

The rule is the basic element of a role. A rule defines what operations the role allows the user to perform. You can apply rules for the following parameters:

Command

A command or group of commands defined in a regular expression.

Feature

Commands that apply to a function provided by the Cisco Nexus device. Enter the **show role feature** command to display the feature names available for this parameter.

Feature group

Default or user-defined group of features. Enter the **show role feature-group** command to display the default feature groups available for this parameter.

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.

You can configure up to 256 rules for each role. The user-specified rule number determines the order in which the rules are applied. Rules are applied in descending order. For example, if a role has three rules, rule 3 is applied before rule 2, which is applied before rule 1.

User Role Policies

You can define user role policies to limit the switch resources that the user can access, or to limit access to interfaces and VLANs.

User role policies are constrained by the rules defined for the role. For example, if you define an interface policy to permit access to specific interfaces, the user does not have access to the interfaces unless you configure a command rule for the role to permit the **interface** command.

If a command rule permits access to specific resources (interfaces, VLANs), the user is permitted to access these resources, even if the user is not listed in the user role policies associated with that user.

User Account Configuration Restrictions

The following words are reserved and cannot be used to configure users:

- adm
- bin

- daemon
- ftp
- ftpuser
- games
- gdm
- gopher
- halt
- lp
- mail
- mailnull
- man
- mtsuser
- news
- nobody
- san-admin
- shutdown
- sync
- sys
- uucp
- xfs

**Caution**

The Cisco Nexus 3600 platform switch does not support all numeric usernames, even if those usernames were created in TACACS+ or RADIUS. If an all numeric username exists on an AAA server and is entered during login, the switch rejects the login request.

User Password Requirements

Cisco Nexus device passwords are case sensitive and can contain alphanumeric characters. Special characters, such as the dollar sign (\$) or the percent sign (%), are not allowed.

**Note**

Beginning with Cisco NX-OS Release 7.2(0)N1(1), special characters, such as the dollar sign (\$) or the percent sign (%), can be used in Cisco Nexus device passwords.



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 than one user account.
- Predefined roles such as network-admin and network-operator are not editable.



Note A user account must have at least one user role.

Configuring User Accounts



Note Changes to user account attributes do not take effect until the user logs in and creates a new session.

You can use any alphanumeric character (or) an _ (underscore) as the first character in a username. Using any other special characters for the first character is not allowed. If the username contains the characters that are not allowed, the specified user is unable to log in.

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

	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 <i>user-id</i> [password <i>password</i>] [expire <i>date</i>] [role <i>role-name</i>]	<p>Configures a user account.</p> <p>The <i>user-id</i> is a case-sensitive, alphanumeric character string with a maximum of 28 characters.</p> <p>The default <i>password</i> is undefined.</p> <p>Note If you do not specify a password, the user might not be able to log into the switch.</p> <p>Note Starting with Release 7.0(3)F3(1), a new internal function is implemented to check the password strength.</p> <p>The expire <i>date</i> option format is YYYY-MM-DD. The default is no expiry date.</p>
Step 4	switch(config) # exit	Exists global configuration mode.
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
```

The following example shows the criteria in enabling the password strength-check starting with Release 7.0(3)F3(1):

```
switch(config)# username xyz password nbv12345
password is weak
Password should contain characters from at least three of the following classes: lower case
letters, upper case letters, digits and special characters.
switch(config)# username xyz password Nbv12345
password is weak
it is too simplistic/systematic
switch(config)#
```

Configuring 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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # role name <i>role-name</i>	Specifies a user role and enters role configuration mode.

	Command or Action	Purpose
		The <i>role-name</i> argument is a case-sensitive, alphanumeric character string with a maximum of 16 characters.
Step 3	<code>switch(config-role) # rule number {deny permit} command command-string</code>	Configures a command rule. The <i>command-string</i> 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	<code>switch(config-role)# rule number {deny permit} {read read-write}</code>	Configures a read-only or read-and-write rule for all operations.
Step 5	<code>switch(config-role)# rule number {deny permit} {read read-write} feature feature-name</code>	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	<code>switch(config-role)# rule number {deny permit} {read read-write} feature-group group-name</code>	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) <code>switch(config-role)# description text</code>	Configures the role description. You can include spaces in the description.
Step 8	<code>switch(config-role)# end</code>	Exits role configuration mode.
Step 9	(Optional) <code>switch# show role</code>	Displays the user role configuration.
Step 10	(Optional) <code>switch# copy running-config startup-config</code>	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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # role feature-group <i>group-name</i>	Specifies a user role feature group and enters role feature group configuration mode. The <i>group-name</i> 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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # role name <i>role-name</i>	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 <i>interface-list</i>	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

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters global configuration mode.
Step 2	switch(config) # role name <i>role-name</i>	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 <i>vlan-list</i>	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 [<i>role-name</i>]	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.

Default Settings for the User Accounts and RBAC

The following table lists the default settings for user accounts and RBAC parameters.

Table 27: 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.



INDEX

A

- action statements [145](#)
 - EEM [145](#)
- action statements, configuring [152](#)
 - EEM [152](#)
- activating sessions [191](#)
 - SPAN [191](#)
- adding show commands, alert groups [86](#)
 - smart call home [86](#)
- additional references [176](#)
 - EEM [176](#)
- agent address [230](#)
 - sFlow [230](#)
- alert groups [73](#)
 - smart call home [73](#)
- analyzer address [229](#)
 - sFlow [229](#)
- analyzer port [230](#)
 - sFlow [230](#)
- associating alert groups [85](#)
 - smart call home [85](#)

C

- call home notifications [92–93](#)
 - full-text format for syslog [92](#)
 - XML format for syslog [93](#)
- clear logging onboard [182](#)
- committing [62](#)
 - NTP configuration changes [62](#)
- configuration example [214, 233](#)
 - ERSPAN [214](#)
 - source [214](#)
 - sFlow [233](#)
- configuration examples [64, 193](#)
 - for SPAN [193](#)
 - NTP [64](#)
- configuration sync after reboot [30](#)
 - switch profiles [30](#)
- configuration, verifying [106](#)
 - scheduler [106](#)
- configure maintenance profile maintenance-mode [238](#)
- configure maintenance profile normal-mode [239](#)

- configuring [51–53, 55–57, 60](#)
 - device as an authoritative NTP server [51](#)
 - NTP authentication [53, 55](#)
 - NTP logging [60](#)
 - NTP server and peer [52](#)
 - NTP source interface [57](#)
 - NTP source IP address [56](#)
- contact information, configuring [81](#)
 - smart call home [81](#)
- counter poll interval [227](#)
 - sFlow [227](#)
- creating, deleting sessions [186](#)
 - SPAN [186](#)

D

- datagram size [228](#)
 - sFlow [228](#)
- default parameters [201](#)
 - ERSPAN [201](#)
- default settings [50, 70, 80, 98, 147, 224](#)
 - EEM [147](#)
 - rollback [70](#)
 - scheduler [98](#)
 - sFlow [224](#)
 - smart call home [80](#)
- default SNMP settings [114](#)
- defining EEM policies [154](#)
 - VSH script [154](#)
- description, configuring [190](#)
 - SPAN [190](#)
- destination ports, characteristics [184](#)
 - SPAN [184](#)
- destination profile, creating [83](#)
 - smart call home [83](#)
- destination profile, modifying [84](#)
 - smart call home [84](#)
- destination profiles [72](#)
 - smart call home [72](#)
- destinations [184](#)
 - SPAN [184](#)
- device IDs [75](#)
 - call home format [75](#)
- diagnostics [139–140, 142](#)
 - configuring [140](#)

- diagnostics (*continued*)
 - default settings [142](#)
 - expansion modules [140](#)
 - health monitoring [140](#)
 - runtime [139](#)
- disabling [106](#)
 - scheduler [106](#)
- discarding [62](#)
 - NTP configuration changes [62](#)
- displaying information [192](#)
 - SPAN [192](#)
- downgrading software [185](#)
 - loss of SPAN configurations [185](#)
- duplicate message throttling, disabling [89–90](#)
 - smart call home [89–90](#)

E

- e-mail details, configuring [87](#)
 - smart call home [87](#)
- e-mail notifications [71](#)
 - smart call home [71](#)
- EEE [146](#)
 - guidelines and limitations [146](#)
- EEM [143–149, 152, 154–155, 157, 176](#)
 - action statements [145](#)
 - action statements, configuring [152](#)
 - additional references [176](#)
 - default settings [147](#)
 - defining environment variables [147](#)
 - event statements [144](#)
 - event statements, configuring [149](#)
 - licensing [146](#)
 - policies [143](#)
 - prerequisites [146](#)
 - syslog script [157](#)
 - system policies, overriding [155](#)
 - user policy, defining [148](#)
 - VSH script [154](#)
 - registering and activating [154](#)
 - VSH script policies [146](#)
- embedded event manager [143](#)
 - overview [143](#)
- enabling [61, 99](#)
 - CFS distribution for NTP [61](#)
 - scheduler [99](#)
- environment variables, defining [147](#)
 - EEM [147](#)
- ERSPAN [197–198, 201, 214, 216](#)
 - configuring source sessions [201](#)
 - default parameters [201](#)
 - high availability [198](#)
 - information about [197](#)
 - prerequisites [198](#)
 - related documents [216](#)

- ERSPAN (*continued*)
 - sessions [198](#)
 - multiple [198](#)
 - source [214](#)
 - configuration example [214](#)
 - source sessions [201](#)
 - configuring for ERSPAN [201](#)
 - sources [197](#)
- Ethernet destination port, configuring [186](#)
 - SPAN [186](#)
- event statements [144](#)
 - EEM [144](#)
- event statements, configuring [149](#)
 - EEM [149](#)
- example [107](#)
 - job schedule, displaying [107](#)
 - scheduler job, creating [107](#)
 - scheduler job, scheduling [107](#)
 - scheduler jobs, displaying results [107](#)
- example, local and peer sync [36](#)
 - switch profiles [36](#)
- executing a session [69](#)

F

- feature groups, creating [284](#)
 - RBAC [284](#)
- filtering SNMP requests [118](#)

G

- GOLD diagnostics [139–140](#)
 - configuring [140](#)
 - expansion modules [140](#)
 - health monitoring [140](#)
 - runtime [139](#)
- guidelines [224](#)
 - sFlow [224](#)
- guidelines and limitations [16, 79, 98, 113, 146, 185, 280](#)
 - EEM [146](#)
 - scheduler [98](#)
 - smart call home [79](#)
 - SNMP [113](#)
 - SPAN [185](#)
 - switch profiles [16](#)
 - user accounts [280](#)
- guidelines and limitations for configuration rollback [265](#)

H

- health monitoring diagnostics [140](#)
 - information [140](#)
- high availability [41](#)
 - PTP [41](#)
 - high availability [41](#)

- hw-module logging onboard 179
- hw-module logging onboard counter-stats 179
- hw-module logging onboard cpuhog 179
- hw-module logging onboard environmental-history 179
- hw-module logging onboard error-stats 180
- hw-module logging onboard interrupt-stats 180
- hw-module logging onboard module 180
- hw-module logging onboard obfl-logs 180

I

- IDs 75
 - serial IDs 75
- information 47
 - ntp 47
- information about 48, 97
 - clock manager 48
 - distributing NTP using CFS 48
 - NTP as time server 48
 - scheduler 97
- interfaces, configuring 44
 - PTP 44
- isolate 236

J

- job schedule, displaying 107
 - example 107
- job, deleting 102
 - scheduler 102

L

- licensing 146
 - EEM 146
- linkDown notifications 125
- linkUp notifications 125
- log file size, defining 99
 - scheduler 99
- log file, clearing 105
 - scheduler 105
- log files 98
 - scheduler 98

M

- message encryption 117
 - SNMP 117

N

- no isolate 236
- no shutdown 236
- no system interface shutdown 236
- no system mode maintenance 246

- no system mode maintenance dont-generate-profile 246
- no system mode maintenance on-reload reset-reason 244
- notification receivers 119
 - SNMP 119
- NTO on an interface, Enabling and disabling 50
- ntp 47–48
 - information 47
 - virtualization 48
- NTP Broadcast Server, Configuring 58
- NTP multicast client, Configuring 60
- NTP multicast server, Configuring 59

O

- overview 143
 - embedded event manager 143

P

- password requirements 279
- periodic inventory notifications, configuring 88
 - smart call home 88
- policies 143
 - EEM 143
- prerequisites 49, 146, 198, 224
 - EEM 146
 - ERSPAN 198
 - NTP 49
 - sFlow 224
- PTP 39–42, 44
 - configuring globally 42
 - default settings 41
 - device types 39
 - interface, configuring 44
 - overview 39
 - process 40
- python instance 237

R

- rate limit, configuring 188
 - SPAN 188
- RBAC 277–278, 281–282, 284–286
 - feature groups, creating 284
 - rules 278
 - user account restrictions 278
 - 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 80
 - smart call home 80

- related documents [216](#)
 - ERSPAN [216](#)
 - releasing [63](#)
 - CSF session lock [63](#)
 - remote user authentication [98](#)
 - scheduler [98](#)
 - remote user authentication, configuring [100–101](#)
 - scheduler [100–101](#)
 - requirements [279](#)
 - user passwords [279](#)
 - roles [277](#)
 - authentication [277](#)
 - rollback [67, 70](#)
 - checkpoint copy [67](#)
 - creating a checkpoint copy [67](#)
 - default settings [70](#)
 - deleting a checkpoint file [67](#)
 - description [67](#)
 - example configuration [67](#)
 - guidelines [67](#)
 - high availability [67](#)
 - implementing a rollback [67](#)
 - limitations [67](#)
 - reverting to checkpoint file [67](#)
 - verifying configuration [70](#)
 - rules [278](#)
 - RBAC [278](#)
 - running config, displaying [34](#)
 - switch profiles [34](#)
 - runtime diagnostics [139](#)
 - information [139](#)
- ## S
- sampling data source [231](#)
 - sFlow [231](#)
 - sampling rate [225](#)
 - sFlow [225](#)
 - scheduler [97–103, 105–106, 98](#)
 - configuration, verifying [106](#)
 - default settings [98](#)
 - disabling [106](#)
 - enabling [99](#)
 - guidelines and limitations [98](#)
 - information about [97](#)
 - job, deleting [102](#)
 - log file size, defining [99](#)
 - log file, clearing [105](#)
 - log files [98](#)
 - remote user authentication [98](#)
 - remote user authentication, configuring [100–101](#)
 - standards [108](#)
 - timetable, defining [103](#)
 - scheduler job, creating [107](#)
 - example [107](#)
 - scheduler job, scheduling [107](#)
 - example [107](#)
 - scheduler jobs, displaying results [107](#)
 - example [107](#)
 - serial IDs [75](#)
 - description [75](#)
 - server IDs [75](#)
 - description [75](#)
 - session manager [67, 69–70](#)
 - committing a session [69](#)
 - configuring an ACL session (example) [69](#)
 - description [67](#)
 - discarding a session [69](#)
 - guidelines [67](#)
 - limitations [67](#)
 - saving a session [69](#)
 - verifying configuration [70](#)
 - verifying the session [69](#)
 - sFlow [223–225, 227–233](#)
 - agent address [230](#)
 - analyzer address [229](#)
 - analyzer port [230](#)
 - configuration example [233](#)
 - counter poll interval [227](#)
 - datagram size [228](#)
 - default settings [224](#)
 - guidelines [224](#)
 - prerequisites [224](#)
 - sampling data source [231](#)
 - sampling rate [225](#)
 - show commands [232](#)
 - show commands [232](#)
 - sFlow [232](#)
 - show interface brief [248](#)
 - show logging onboard [180](#)
 - show logging onboard boot-uptime [181](#)
 - show logging onboard counter-stats [181](#)
 - show logging onboard credit-loss [181](#)
 - show logging onboard device-version [181](#)
 - show logging onboard endtime [181](#)
 - show logging onboard environmental-history [181](#)
 - show logging onboard error-stats [181](#)
 - show logging onboard exception-log [181](#)
 - show logging onboard interrupt-stats [181](#)
 - show logging onboard module [181](#)
 - show logging onboard obfl-history [181](#)
 - show logging onboard obfl-logs [181](#)
 - show logging onboard stack-trace [181](#)
 - show logging onboard starttime [181](#)
 - show logging onboard status [181](#)
 - show maintenance on-reload reset-reasons [248](#)
 - show maintenance profile [248](#)
 - show maintenance profile maintenance-mode [238, 248](#)
 - show maintenance profile normal-mode [239, 248](#)
 - show maintenance timeout [248](#)
 - show running-config mmode [248](#)

- show snapshots [241, 248](#)
- show snapshots compare [241, 248](#)
- show snapshots dump [249](#)
- show snapshots sections [249](#)
- show startup-config mmode [248](#)
- show system mode [245, 247, 249](#)
- shutdown [236](#)
- sleep instance [236](#)
- smart call home [71–73, 79–81, 83–92](#)
 - adding show commands, alert groups [86](#)
 - alert groups [73](#)
 - associating alert groups [85](#)
 - contact information, configuring [81](#)
 - default settings [80](#)
 - description [71](#)
 - destination profile, creating [83](#)
 - destination profile, modifying [84](#)
 - destination profiles [72](#)
 - duplicate message throttling, disabling [89–90](#)
 - e-mail details, configuring [87](#)
 - guidelines and limitations [79](#)
 - message format options [72](#)
 - periodic inventory notifications [88](#)
 - prerequisites [79](#)
 - registering [80](#)
 - testing the configuration [91](#)
 - verifying [92](#)
- smart call home messages [72, 74](#)
 - configuring levels [74](#)
 - format options [72](#)
- snapshot create [240](#)
- snapshot delete [240](#)
- SNMP [109–115, 117–119, 121, 127–128](#)
 - access groups [113](#)
 - configuring local engineID [127](#)
 - configuring users [115](#)
 - default settings [114](#)
 - disabling [128](#)
 - filtering requests [118](#)
 - functional overview [109](#)
 - group-based access [113](#)
 - guidelines and limitations [113](#)
 - inband access [121](#)
 - message encryption [117](#)
 - notification receivers [119](#)
 - security model [111](#)
 - trap notifications [110](#)
 - user synchronization with CLI [112](#)
 - user-based security [111](#)
 - SNMP [111](#)
 - version 3 security features [110](#)
- SNMP (Simple Network Management Protocol) [110](#)
 - versions [110](#)
- SNMP notification receivers [120](#)
 - configuring with VRFs [120](#)
- SNMP notifications [121](#)
 - filtering based on a VRF [121](#)
- snmp-server name [116](#)
- SNMPv3 [110, 118](#)
 - assigning multiple roles [118](#)
 - security features [110](#)
- software [185](#)
 - downgrading [185](#)
 - loss of SPAN configurations [185](#)
- source IDs [75](#)
 - call home event format [75](#)
- source ports, characteristics [184](#)
 - SPAN [184](#)
- source ports, configuring [188](#)
 - SPAN [188](#)
- SPAN [183–186, 188–193](#)
 - activating sessions [191](#)
 - characteristics, source ports [184](#)
 - configuration examples [193](#)
 - configuration loss when downgrading software [185](#)
 - creating, deleting sessions [186](#)
 - description, configuring [190](#)
 - destination ports, characteristics [184](#)
 - destinations [184](#)
 - displaying information [192](#)
 - egress sources [183](#)
 - Ethernet destination port, configuring [186](#)
 - guidelines and limitations [185](#)
 - ingress sources [183](#)
 - rate limit, configuring [188](#)
 - source port channels, configuring [189](#)
 - source ports, configuring [188](#)
 - sources for monitoring [183](#)
 - VLANs, configuring [189](#)
- SPAN sources [183](#)
 - egress [183](#)
 - ingress [183](#)
- standards [108](#)
 - scheduler [108](#)
- switch profile buffer, displaying [29, 36](#)
- switch profiles [16, 29–30, 34–36](#)
 - buffer, displaying [29, 36](#)
 - configuration sync after reboot [30](#)
 - example, local and peer sync [34, 36](#)
 - guidelines and limitations [16](#)
 - running config, displaying [34](#)
 - verify and commit, displaying [35](#)
- Switched Port Analyzer [183](#)
- syslog [157](#)
 - EEM [157](#)
- system interface shutdown [236](#)
- system mode maintenance dont-generate-profile [244](#)
- system mode maintenance on-reload reset-reason [244](#)
- system mode maintenance shutdown [244](#)
- system mode maintenance timeout [244](#)

system policies, overriding [155](#)
EEM [155](#)

T

testing the configuration [91](#)
smart call home [91](#)
timetable, defining [103](#)
scheduler [103](#)
trap notifications [110](#)

U

user account restrictions [278](#)
RBAC [278](#)
user accounts [279–280, 286](#)
guidelines and limitations [280](#)
passwords [279](#)
verifying [286](#)
user policies, defining [148](#)
EEM [148](#)
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 [63, 92, 286](#)
NTP configuration [63](#)
RBAC [286](#)
smart call home [92](#)
user accounts [286](#)
virtualization [48](#)
ntp [48](#)
VRFs [120–121](#)
configuring SNMP notification receivers with [120](#)
filtering SNMP notifications [121](#)
VSH script [154](#)
defining EEM policies [154](#)
VSH script policies [146, 154](#)
EEM [146](#)
registering and activating [154](#)