Cisco Nexus 3600 NX-OS Fundamentals Configuration Guide, Release 10.1(x)

First Published: 2021-02-16

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

CONTENTS

Preface

This preface includes the following sections:

Audience

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

Document Conventions

Command descriptions use the following conventions:

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

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

Examples use the following conventions:

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

Related Documentation for Cisco Nexus 3000 Series Switches

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

https://www.cisco.com/c/en/us/support/switches/nexus-3000-series-switches/tsd-products-support-series-home.html

Documentation Feedback

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

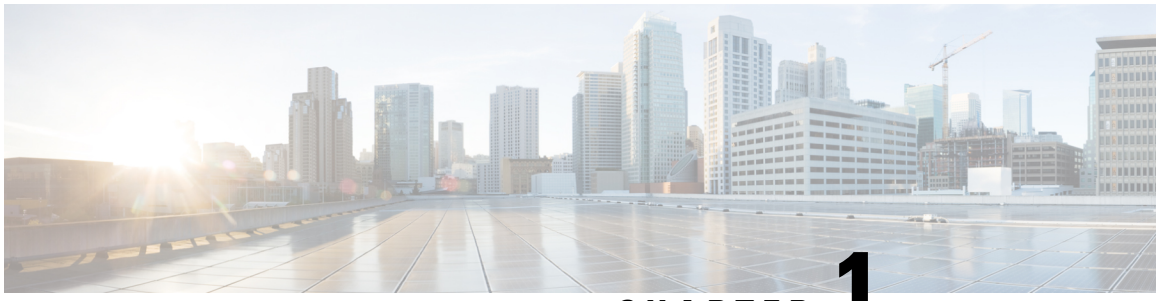
Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business 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.

Preface



CHAPTER 1

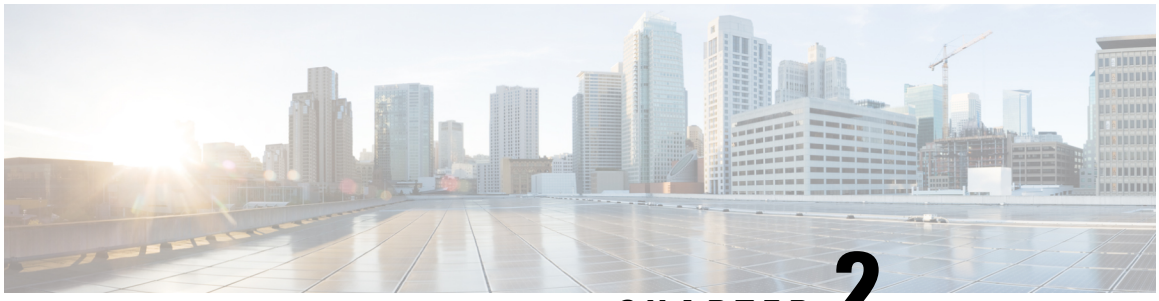
New and Changed Information

New and Changed Information

Table 1: New and Changed Features for Cisco NX-OS Release 10.1(x)

Feature	Description	Changed in Release	Where Documented
There are no new features for this release.		10.1(1)	

New and Changed Information



CHAPTER 2

Understanding the Command-Line Interface

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*.

Information About the CLI Prompt

Once you have successfully accessed the device, the CLI prompt displays in the terminal window of your console port or remote workstation as shown in the following example:

```
User Access Verification
login: admin
Password:<password>
Cisco Nexus Operating System (NX-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2009, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained in this software are
owned by other third parties and used and distributed under
license. Certain components of this software are licensed under
the GNU General Public License (GPL) version 2.0 or the GNU
Lesser General Public License (LGPL) Version 2.1. A copy of each
such license is available at
http://www.opensource.org/licenses/gpl-2.0.php and
http://www.opensource.org/licenses/lgpl-2.1.php
```

```
switch#
```

You can change the default device hostname.

From the CLI prompt, you can do the following:

- Use CLI commands for configuring features
- Access the command history
- Use command parsing functions

Note In normal operation, usernames are case sensitive. However, when you are connected to the device through its console port, you can enter a login username in all uppercase letters regardless of how the username was defined. As long as you provide the correct password, the device logs you in.

Command Modes

This section describes command modes in the Cisco NX-OS CLI.

Special Characters

This table lists the characters that have special meaning in Cisco NX-OS text strings and should be used only in regular expressions or other special contexts.

Table 2: Special Characters

Character	Description
%	Percent
#	Pound, hash, or number
...	Ellipsis
|	Vertical bar
< >	Less than or greater than
[ ]	Brackets
{ }	Braces

Keystroke Shortcuts

This table lists command key combinations that can be used in both EXEC and configuration modes.

Table 3: Keystroke Shortcuts

Keystokes	Description
Ctrl-A	Moves the cursor to the beginning of the line.
Ctrl-B	Moves the cursor one character to the left. When you enter a command that extends beyond a single line, you can press the Left Arrow or Ctrl-B keys repeatedly to scroll back toward the system prompt and verify the beginning of the command entry, or you can press the Ctrl-A key combination.
Ctrl-C	Cancels the command and returns to the command prompt.
Ctrl-D	Deletes the character at the cursor.
Ctrl-E	Moves the cursor to the end of the line.
Ctrl-F	Moves the cursor one character to the right.
Ctrl-G	Exits to the previous command mode without removing the command string.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-L	Redisplays the current command line.
Ctrl-N	Displays the next command in the command history.
Ctrl-O	Clears the terminal screen.
Ctrl-P	Displays the previous command in the command history.
Ctrl-R	Redisplays the current command line.
Ctrl-T	Transposes the character under the cursor with the character located to the right of the cursor. The cursor is then moved one character to the right.
Ctrl-U	Deletes all characters from the cursor to the beginning of the command line.
Ctrl-V	Removes any special meaning for the following keystroke. For example, press Ctrl-V before entering a question mark (?) in a regular expression.
Ctrl-W	Deletes the word to the left of the cursor.
Ctrl-X, H	Lists the history of commands you have entered. When using this key combination, press and release the Ctrl and X keys together before pressing H.
Ctrl-Y	Recalls the most recent entry in the buffer (press keys simultaneously).

Keystokes	Description
Ctrl-Z	Ends a configuration session, and returns you to EXEC mode. When used at the end of a command line in which a valid command has been typed, the resulting configuration is first added to the running configuration file.
Up arrow key	Displays the previous command in the command history.
Down arrow key	Displays the next command in the command history.
Right arrow key Left arrow key	Moves your cursor through the command string, either forward or backward, allowing you to edit the current command.
?	Displays a list of available commands.

Keystokes	Description

Tab	Completes the word for you after you enter the first characters of the word and then press the Tab key. All options that match are presented.
	Use tabs to complete the following items:
	• Command names
	• Scheme names in the file system
	• Server names in the file system
	• Filenames in the file system
	Example:
	`switch(config)# xm<Tab>`
	`switch(config)# xml<Tab>`
	`switch(config)# xml server`

	Example:
	`switch(config)# c<Tab>`
	`callhome  class-map  clock  cts`
	`cdp  cli  control-plane`
	`switch(config)# cl<Tab>`
	`class-map  cli  clock`
	`switch(config)# cla<Tab>`
	`switch(config)# class-map`

	Example:
	`switch# cd bootflash:<Tab>`
	`bootflash:  bootflash://sup-1/`
	`bootflash:///  bootflash://sup-2/`
	`bootflash://module-5/  bootflash://sup-active/`
	`bootflash://module-6/  bootflash://sup-local/`

	Example:
	`switch# cd bootflash://mo<Tab>`
	`bootflash://module-5/  bootflash://module-6/cv`
	`switch# cd bootflash://module-`

Abbreviating Commands

You can abbreviate commands and keywords by entering the first few characters of a command. The abbreviation must include sufficient characters to make it unique from other commands or keywords. If you are having trouble entering a command, check the system prompt and enter the question mark (?) for a list of available commands. You might be in the wrong command mode or using incorrect syntax.

This table lists examples of command abbreviations.

Table 4: Examples of Command Abbreviations

Command	Abbreviation
configure terminal	**conf t**
copy running-config startup-config	**copy run start**
interface ethernet 1/2	**int e 1/2**
show running-config	**sh run**

Completing a Partial Command Name

If you cannot remember a complete command name, or if you want to reduce the amount of typing you have to perform, enter the first few letters of the command, and then press the **Tab** key. The command line parser will complete the command if the string entered is unique to the command mode. If your keyboard does not have a **Tab** key, press **Ctrl-I** instead.

The CLI recognizes a command once you have entered enough characters to make the command unique. For example, if you enter **conf** in EXEC mode, the CLI will be able to associate your entry with the **configure** command, because only the **configure** command begins with **conf**.

In the following example, the CLI recognizes the unique string for **conf** in EXEC mode when you press the **Tab** key:

```
switch# conf<Tab>
switch# configure
```

When you use the command completion feature the CLI displays the full command name. The CLI does not execute the command until you press the **Return** or **Enter** key. This feature allows you to modify the command if the full command was not what you intended by the abbreviation. If you enter a set of characters that could indicate more than one command, a list of matching commands displays.

For example, entering **co**<**Tab**> lists all commands available in EXEC mode beginning with **co**:

```
switch# co<Tab>
configure   copy
switch# co
```

Note that the characters you entered appear at the prompt again to allow you to complete the command entry.

Identifying Your Location in the Command Hierarchy

Some features have a configuration submode hierarchy nested more than one level. In these cases, you can display information about your present working context (PWC).

SUMMARY STEPS

1. **where detail**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**where detail**
	Example:
	`switch# configure terminal`
	`switch(config)# interface mgmt0`
	`switch(config-if)# where detail`
	`mode:  conf`
	`  interface mgmt0`
	`  username:  admin`
		Displays the PWC.

Using the no Form of a Command

Almost every configuration command has a **no** form that can be used to disable a feature, revert to a default value, or remove a configuration. The Cisco NX-OS command reference publications describe the function of the **no** form of the command whenever a **no** form is available.

This example shows how to disable a feature:

```
switch# configure terminal
switch(config)# feature tacacs+
switch(config)# no feature tacacs+
```

This example shows how to revert to the default value for a feature:

```
switch# configure terminal
switch(config)# banner motd #Welcome to the switch#
switch(config)# show banner motd
Welcome to the switch

switch(config)# no banner motd
switch(config)# show banner motd
User Access Verification
```

This example shows how to remove the configuration for a feature:

```
switch# configure terminal
switch(config)# radius-server host 10.10.2.2
switch(config)# show radius-server
retransmission count:0
timeout value:1
deadtime value:1
total number of servers:1

following RADIUS servers are configured:
        10.10.1.1:
                available for authentication on port:1812
                available for accounting on port:1813
        10.10.2.2:
                available for authentication on port:1812
                available for accounting on port:1813

switch(config)# no radius-server host 10.10.2.2
switch(config)# show radius-server
```

```
retransmission count:0
timeout value:1
deadtime value:1
total number of servers:1

following RADIUS servers are configured:
        10.10.1.1:
                available for authentication on port:1812
                available for accounting on port:1813
```

This example shows how to use the **no** form of a command in EXEC mode:

```
switch# cli var name testinterface ethernet1/2
switch# show cli variables
SWITCHNAME="switch"
TIMESTAMP="2009-05-12-13.43.13"
testinterface="ethernet1/2"

switch# cli no var name testinterface
switch# show cli variables
SWITCHNAME="switch"
TIMESTAMP="2009-05-12-13.43.13"
```

Configuring CLI Variables

This section describes CLI variables in the Cisco NX-OS CLI.

Command Aliases

This section provides information about command aliases.

Command Scripts

This section describes how you can create scripts of commands to perform multiple tasks.

Context-Sensitive Help

The Cisco NX-OS software provides context-sensitive help in the CLI. You can use a question mark (?) at any point in a command to list the valid input options.

CLI uses the caret (^) symbol to isolate input errors. The ^ symbol appears at the point in the command string where you have entered an incorrect command, keyword, or argument.

This table shows example outputs of context sensitive help.

Table 5: Context-Sensitive Help Example

Example Outputs	Description

`switch# clock ?`
`  set  HH:MM:SS Current Time`
`switch# clock`
	Displays the command syntax for the **clock** command in EXEC mode.
	The switch output shows that the **set** keyword is required for using the **clock** command.

`switch# clock set ?`
`  WORD  HH:MM:SS Current Time`
`switch# clock set`
	Displays the command syntax for setting the time.
	The help output shows that the current time is required for setting the clock and how to format the time.

`switch# clock set 13:32:00<CR>`
`% Incomplete command`
`switch#`
	Adds the current time.
	The CLI indicates the command is incomplete.

`switch# <Ctrl-P>` `switch# clock set 13:32:00`	Displays the previous command that you entered.

`switch# clock set 13:32:00 ?`
`  <1-31>  Day of the month`
`switch# clock set 13:32:00`
	Displays the additional arguments for the **clock set** command.

`switch# clock set 13:32:00 18 ?`
`  April  Month of the year`
`  August  Month of the year`
`  December  Month of the year`
`  February  Month of the year`
`  January  Month of the year`
`  July  Month of the year`
`  June  Month of the year`
`  March  Month of the year`
`  May  Month of the year`
`  November  Month of the year`
`  October  Month of the year`
`  September  Month of the year`
`switch# clock set 13:32:00 18`
	Displays the additional arguments for the **clock set** command.

`switch# clock set 13:32:00 18 April 08<CR>` `% Invalid input detected at '^' marker.`	Adds the date to the clock setting. The CLI indicates an error with the caret symbol (^) at 08.

`switch# clock set 13:32:00 18 April ?`
`  <2000-2030>  Enter the year (no abbreviation)`
 	
`switch# clock set 13:32:00 18 April`
	Displays the correct arguments for the year.

`switch# clock set 13:32:00 18 April 2008<CR>` `switch#`	Enters the correct syntax for the **clock set** command.

Understanding Regular Expressions

The Cisco NX-OS software supports regular expressions for searching and filtering in CLI output, such as the **show** commands. Regular expressions are case sensitive and allow for complex matching requirements.

Searching and Filtering show Command Output

Often, the output from **show** commands can be lengthy and cumbersome. The Cisco NX-OS software provides the means to search and filter the output so that you can easily locate information. The searching and filtering options follow a pipe character (|) at the end of the **show** command. You can display the options using the CLI context-sensitive help facility:

```
switch# show running-config | ?
  cut      Print selected parts of lines.
  diff     Show difference between current and previous invocation (creates temp files:
           remove them with 'diff-clean' command and don't use it on commands with big
           outputs, like 'show tech'!)
  egrep    Egrep - print lines matching a pattern
  grep     Grep - print lines matching a pattern
  head     Display first lines
  human    Output in human format
  last     Display last lines
  less     Filter for paging
  no-more  Turn-off pagination for command output
  perl     Use perl script to filter output
  section  Show lines that include the pattern as well as the subsequent lines that are
           more indented than matching line
  sed      Stream Editor
  sort     Stream Sorter
  sscp     Stream SCP (secure copy)
  tr       Translate, squeeze, and/or delete characters
  uniq     Discard all but one of successive identical lines
  vsh      The shell that understands cli command
  wc       Count words, lines, characters
  xml      Output in xml format (according to .xsd definitions)
  begin    Begin with the line that matches
  count    Count number of lines
  end      End with the line that matches
  exclude  Exclude lines that match
  include  Include lines that match
```

Searching and Filtering from the --More-- Prompt

You can search and filter output from `--More--` prompts in the **show** command output.

This table describes the `--More--` prompt commands.

Table 6: --More-- Prompt Commands

Commands	Description
[*lines*]<space>	Displays output lines for either the specified number of lines or the current screen size.

Commands	Description
[*lines*]**z**	Displays output lines for either the specified number of lines or the current screen size. If you use the *lines* argument, that value becomes the new default screen size.
[*lines*]<return>	Displays output lines for either the specified number of lines or the current default number of lines. The initial default is 1 line. If you use the optional *lines* argument, that value becomes the new default number of lines to display for this command.
[*lines*]**d** or [*lines*]Ctrl+shift+D	Scrolls through output lines for either the specified number of lines or the current default number of lines. The initial default is 11 lines. If you use the optional *lines* argument, that value becomes the new default number of lines to display for this command.
q or **Q** or Ctrl-C	Exits the `--More--` prompt.
[*lines*]**s**	Skips forward in the output for either the specified number of lines or the current default number of lines and displays a screen of lines. The default is 1 line.
[*lines*]**f**	Skips forward in the output for either the specified number of screens or the current default number of screens and displays a screen of lines. The default is 1 screen.
=	Displays the current line number.
[*count*]/*expression*	Skips to the line that matches the regular expression and displays a screen of output lines. Use the optional *count* argument to search for lines with multiple occurrences of the expression. This command sets the current regular expression that you can use in other commands.
[*count*]**n**	Skips to the next line that matches the current regular expression and displays a screen of output lines. Use the optional *count* argument to skip past matches.
{**!** | **:!**[*shell-cmd*]}	Executes the command specified in the *shell-cmd* argument in a subshell.
.	Repeats the previous command.

Using the Command History

The Cisco NX-OS software CLI allows you to access the command history for the current user session. You can recall and reissue commands, with or without modification. You can also clear the command history.

Enabling or Disabling the CLI Confirmation Prompts

For many features, the Cisco NX-OS software displays prompts on the CLI that ask for confirmation before continuing. You can enable or disable these prompts. The default is enabled.

SUMMARY STEPS

1. [**no**] **terminal dont-ask** [**persist**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	[**no**] **terminal dont-ask** [**persist**] **Example:** `switch# terminal dont-ask`	Disables the CLI confirmation prompt. The **persist** keyword makes the setting persistent across sessions for the current username. The default is enabled. Use the **no** form of the command to enable the CLI confirmation prompts.

Setting CLI Display Colors

You can change the CLI colors to display as follows:

- The prompt displays in green if the previous command succeeded.
- The prompt displays in red of the previous command failed.
- The user input displays in blue.
- The command output displays in the default color.

The default colors are those set by the terminal emulator software.

SUMMARY STEPS

1. **terminal color** [**evening**] [**persist**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal color** [**evening**] [**persist**] **Example:** `switch# terminal color`	Sets the CLI display colors for the terminal session. The **evening** keyword is not supported. The **persist** keyword makes the setting persistent across sessions for the current username. The default setting is not persistent.

Sending Commands to Modules

You can send commands directly to modules from the supervisor module session using the **slot** command.

The **slot** has the following syntax:

slot *slot-number* [**quoted**] *command-string*

By default, the keyword and arguments in the *command-string* argument are separated by a space. To send more than one command to a module, separate the commands with a space character, a semicolon character (;), and a space character.

The **quoted** keyword indicates that the command string begins and ends with double quotation marks ("). Use this keyword when you want to redirect the module command output to a filtering utility, such as diff, that is supported only on the supervisor module session.

The following example shows how to display and filter module information:

```
switch# slot 1 show version | grep lc
```

This example shows how to filter module information on the supervisor module session:

```
switch# slot 1 quoted "show version" | diff
```

BIOS Loader Prompt

When the supervisor modules power up, a specialized BIOS image automatically loads and tries to locate a valid kickstart image for booting the system. If a valid kickstart image is not found, the following BIOS loader prompt displays:

```
loader>
```

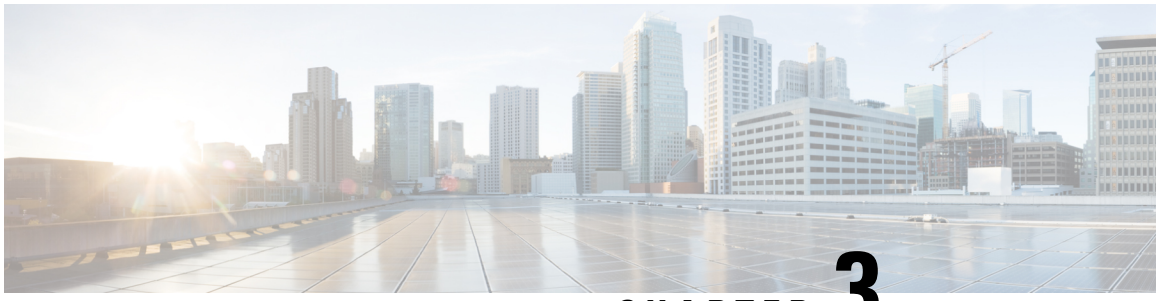
For information on how to load the Cisco NX-OS software from the `<loader>` prompt, see the Cisco Nexus troubleshooting guide for your device.

Examples Using the CLI

This section includes examples of using the CLI.

Additional References for the CLI

This section includes additional information related to the CLI.



CHAPTER 3

Configuring Terminal Settings and Sessions

Information About Terminal Settings and Sessions

This section includes information about terminal settings and sessions.

Configuring the Console Port

You can set the following characteristics for the console port:

- Data bits
- Inactive session timeout
- Parity
- Speed
- Stop bits

Before you begin

Log in to the console port.

SUMMARY STEPS

1. **configure terminal**

2. **line console**
3. **databits** *bits*
4. **exec-timeout** *minutes*
5. **parity** {**even** | **none** | **odd**}
6. **speed** {**300** | **1200** | **2400** | **4800** | **9600** | **38400** | **57600** | **115200**}
7. **stopbits** {**1** | **2**}
8. **exit**
9. (Optional) **show line console**
10. (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	**line console**
	Example:
	`switch# line console`
	`switch(config-console)#`
		Enters console configuration mode.

Step 3	**databits** *bits* **Example:** `switch(config-console)# databits 7`	Configures the number of data bits per byte. The range is from 5 to 8. The default is 8.
Step 4	**exec-timeout** *minutes* **Example:** `switch(config-console)# exec-timeout 30`	Configures the timeout for an inactive session. The range is from 0 to 525600 minutes (8760 hours). A value of 0 minutes disables the session timeout. The default is 30 minutes.
Step 5	**parity** {**even**	**none**	**odd**} **Example:** `switch(config-console)# parity even`	Configures the parity. The default is **none**.
Step 6	**speed** {**300**	**1200**	**2400**	**4800**	**9600**	**38400**	**57600**	**115200**} **Example:** `switch(config-console)# speed 115200`	Configures the transmit and receive speed. The default is **115200**.
Step 7	**stopbits** {**1**	**2**} **Example:** `switch(config-console)# stopbits 2`	Configures the stop bits. The default is **1**.
Step 8	**exit** **Example:**	Exits console configuration mode.

	Command or Action	Purpose
	`switch(config-console)# exit` `switch(config)#`	
Step 9	(Optional) **show line console** **Example:** `switch(config)# show line console`	Displays the console settings.
Step 10	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring Virtual Terminals

This section describes how to configure virtual terminals on Cisco NX-OS devices.

Configuring Modem Connections

You can connect a modem to the console port.

Clearing Terminal Sessions

You can clear terminal sessions on the Cisco NX-OS device.

SUMMARY STEPS

1. (Optional) **show users**
2. **clear line** *name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **show users** **Example:** `switch# show users`	Displays the user sessions on the device.
Step 2	**clear line** *name* **Example:** `switch# clear line pts/0`	Clears a terminal session on a specific line. The line name is case sensitive.

Displaying Terminal and Session Information

To display terminal and session information, perform one of the following tasks:

Command	Purpose
show terminal	Displays terminal settings.
show line	Displays the console ports settings.
show users	Displays virtual terminal sessions.
show running-config [**all**]	Displays the user account configuration in the running configuration. The **all** keyword displays the default values for the user accounts.

For detailed information about the fields in the output from these commands, see the Cisco Nexus command reference guide for your device.

Default Settings for File System Parameters

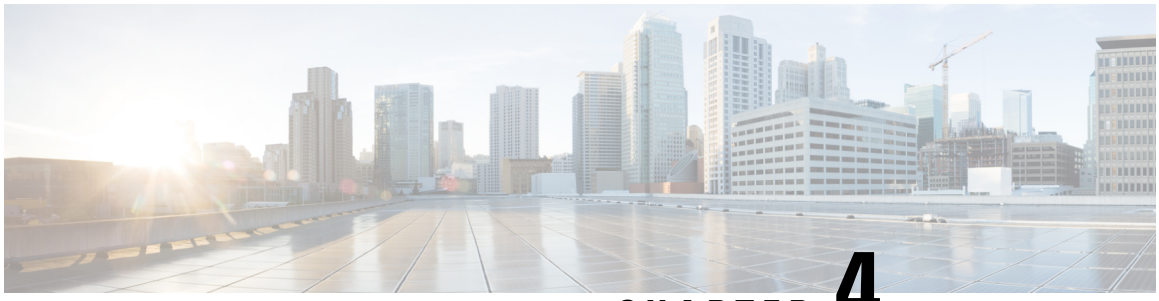
This table lists the default settings for the file system parameters.

Table 7: Default File System Settings

Parameters	Default
Default filesystem	bootflash:

Additional References for Terminal Settings and Sessions

This section includes additional references for terminal settings and sessions on NX-OS devices.



CHAPTER 4

Basic Device Management

Information About Basic Device Management

This section provides information about basic device management.

Guidelines for Password Recovery

Follow these guidelines to recover the password:

- You must be logged in as admin to change the admin password.
- For Cisco Nexus 36180YC-R chassis, press Ctrl-L to interrupt the boot process and get the >loader prompt.

Changing the Device Hostname

You can change the device hostname displayed in the command prompt from the default (switch) to another character string.

SUMMARY STEPS

1. **configure terminal**
2. {**hostname** | **switchname**} *name*
3. **exit**
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	{**hostname** | **switchname**} *name*
	Example:
	Using the **hostname** command:
	`switch(config)# hostname Engineering1`
	`Engineering1(config)#`
	Using the **switchname** command:
	`Engineering1(config)# switchname Engineering2`
	`Engineering2(config)#`
		Changes the device hostname. The *name* argument is alphanumeric, case sensitive, and has a maximum length of 32 characters. The default is switch.
		Note The **switchname** command performs the same function as the **hostname** command.

Step 3	**exit**
	Example:
	`Engineering2(config)# exit`
	`Engineering2#`
		Exits global configuration mode.

Step 4	(Optional) **copy running-config startup-config** **Example:** `Engineering2# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring the MOTD Banner

You can configure the MOTD to display before the login prompt on the terminal when a user logs in. The MOTD banner has the following characteristics:

- Maximum of 80 characters per line
- Maximum of 40 lines

SUMMARY STEPS

1. **configure terminal**
2. **banner motd** *delimiting-character message delimiting-character*
3. **exit**

4. (Optional) **show banner motd**
5. (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	**banner motd** *delimiting-character message delimiting-character*
	Example:
	`switch(config)# banner motd #Welcome to the Switch#`
	`switch(config)#`
		Configures the MOTD banner. Do not use the *delimiting-character* in the *message* text.
		Note Do not use " or % as a delimiting character.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show banner motd** **Example:** `switch# show banner motd`	Displays the configured MOTD banner.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring the Time Zone

You can configure the time zone to offset the device clock time from UTC.

SUMMARY STEPS

1. **configure terminal**
2. **clock timezone** *zone-name offset-hours offset-minutes*
3. **exit**
4. (Optional) **show clock**
5. (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	**clock timezone** *zone-name offset-hours offset-minutes* **Example:** `switch(config)# clock timezone EST -5 0`	Configures the time zone. The *zone-name* argument is a 3-character string for the time zone acronym (for example, PST or EST). The *offset-hours* argument is the offset from the UTC and the range is from –23 to 23 hours. The range for the *offset-minutes* argument is from 0 to 59 minutes.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show clock** **Example:** `switch# show clock`	Displays the time and time zone.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Configuring Summer Time (Daylight Saving Time)

You can configure when summer time, or daylight saving time, is in effect for the device and the offset in minutes.

SUMMARY STEPS

1. **configure terminal**
2. **clock summer-time** *zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes*
3. **exit**
4. (Optional) **show clock detail**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

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

	Command or Action	Purpose
	`switch# configure terminal` `switch(config)#`	
Step 2	**clock summer-time** *zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes* **Example:** `switch(config)# clock summer-time PDT 1 Sunday March 02:00 1 Sunday November 02:00 60`	Configures summer time or daylight saving time. The *zone-name* argument is a three character string for the time zone acronym (for example, PST and EST). The values for the *start-day* and *end-day* arguments are **Monday**, **Tuesday**, **Wednesday**, **Thursday**, **Friday**, **Saturday**, and **Sunday**. The values for the *start-month* and *end-month* arguments are **January**, **February**, **March**, **April**, **May**, **June**, **July**, **August**, **September**, **October**, **November**, and **December**. The value for the *start-time* and *end-time* arguments are in the format *hh***:***mm*. The range for the *offset-minutes* argument is from 0 to 1440 minutes.

Step 3	**exit**
	Example:
	`switch(config)# exit`
	`switch#`
		Exits global configuration mode.

Step 4	(Optional) **show clock detail** **Example:** `switch(config)# show clock detail`	Displays the configured MOTD banner.
Step 5	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Manually Setting the Device Clock

You can set the clock manually if your device cannot access a remote time source.

Before you begin

Configure the time zone.

SUMMARY STEPS

1. **clock set** *time day month year*
2. (Optional) **show clock**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**clock set** *time day month year*
	Example:
	`switch# clock set 15:00:00 30 May 2008`
	`Fri May 30 15:14:00 PDT 2008`
		Configures the device clock.
		The format for the *time* argument is *hh*:*mm*:*ss*.
		The range for the *day* argument is from 1 to 31.
		The values for the *month* argument are **January**, **February**, **March**, **April**, **May**, **June**, **July**, **August**, **September**, **October**, **November**, and **December**.
		The range for the *year* argument is from 2000 to 2030.

Step 2	(Optional) **show clock** **Example:** `switch(config)# show clock`	Displays the current clock value.

Related Topics

Configuring the Time Zone, on page 23

Setting the Clock Manager

You can configure the clock manager to synchronize all the clocks of the components in the Cisco Nexus chassis.

SUMMARY STEPS

1. **clock protocol** *protocol* **vdc** *vdc-num*
2. (Optional) **show run clock_manager**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**clock protocol** *protocol* **vdc** *vdc-num* **Example:** `# clock protocol ptp vdc 2`	Configures the clock manager. The values for the *protocol* argument are **ptp**, **ntp**, and **none**. The following describes the values: • **ptp**—Synchronizes clocks with Precision Time Protocol (PTP) as described by IEEE 1588. • **ntp**— Synchronizes clocks with Network Time Protocol (NTP). • **none**—Use **clock set** to set supervisor clocks. **Note** When **none** is used, the clock in the specified VDC must be configured.

	Command or Action	Purpose
		Note Once the protocol is configured, the clock in the specified VDC must use that protocol. For example, if the **clock protocol ptp vdc 2** command is entered, then PTP should be configured in VDC 2. The range for the *vdc* argument is 1 to 8.
Step 2	(Optional) **show run clock_manager** **Example:** `#show run clock_manager`	Displays the configuration of the clock manager.

Managing Users

You can display information about users logged into the device and send messages to those users.

Verifying the Device Configuration

To verify the configuration after bootstrapping the device using POAP, use one of the following commands:

Command	Purpose
show running-config	Displays the running configuration.
show startup-config	Displays the startup configuration.
show time-stamp running-config last-changed	Displays the timestamp when the running configuration was last changed.

For detailed information about the fields in the output from these commands, see the Cisco Nexus command reference for your device.

Default Settings for Basic Device Parameters

This table lists the default settings for basic device parameters.

Table 8: Default Basic Device Parameters

Parameters	Default
MOTD banner text	User Access Verification
Clock time zone	UTC

Consistency Checker Commands

Cisco NX-OS provides consistency checker commands to validate the software state with the hardware state. The result of the consistency checker is logged as either PASSED or FAILED.

```
2019 May 1 16:31:39 switch vshd: CC_LINK_STATE:
Consistency Check: PASSED
```

Consistency checker is a tool that performs the following functions:

- Checks for system consistency
- Helps perform root cause analysis and fault isolation
- Checks for consistency between software and hardware tables

Cisco NX-OS supports the following consistency checker commands.

Table 9: Consistency Checker Commands

Command	Description
show consistency-checker copp	Verifies CoPP programming.
show consistency-checker egress-xlate private-vlan	Verifies the private VLAN egress-xlate in the hardware.
show consistency-checker forwarding single-route {**ipv4** | **ipv6**} *ip-address* **vrf** *vrf-name*} [**brief** | **detail**]	Checks for Layer 3 route consistency for a specific route.
show consistency-checker kim interface {**ethernet** *slot/port* | **port-channel** *number* | **vlan** *vlan-id*} [**brief** | **detail**]	Verifies the internal connectivity between the supervisor and the line card.
show consistency-checker l2 module *module-number*	Verifies that learned MAC addresses are consistent between the software and the hardware. It also shows extra entries that are present in the hardware but not in the software and missing entries in the hardware.
show consistency-checker l3-interface interface ethernet *slot/port* [**brief** | **detail**]	Checks for Layer 3 settings of an interface in the hardware and for the following configuration in the hardware: L3 VLAN, CML Flags, IPv4 Enable, VPN ID. This command works for physical interfaces and interfaces that are part of a port channel. It does not validate subinterfaces or FEX interfaces.
show consistency-checker l3-interface module *module-number* [**brief** | **detail**]	Checks for Layer 3 settings of all interfaces in the module and for the following configuration in the hardware: L3 VLAN, CML Flags, IPv4 Enable, VPN ID. This command works for physical interfaces and interfaces that are part of a port channel. It does not validate subinterfaces.

Command	Description
show consistency-checker link-state module *module-number* [**brief** | **detail**]	Verifies the software link state of all the interfaces in the module against its hardware link state. This command works for physical Ethernet interfaces and physical Ethernet interfaces that are part of a port channel. It does not validate subinterfaces or FEX interfaces.
show consistency-checker membership port-channels [**interface port-channel** *channel-number*] [**brief** | **detail**]	Checks for port-channel membership in the hardware in all modules and validates it with the software state. This command runs per port channel.
show consistency-checker membership port-channels [**brief** | **detail**]	Checks for port-channel membership in the hardware in all modules and validates it with the software state. This command runs for all port channels in the system.
show consistency-checker membership vlan *vlan-id* {**native-vlan** | **private-vlan interface** {**ethernet** *slot/port* | **port-channel** *number* | **native-vlan**}} [**brief** | **detail**]	Determines that the VLAN membership in the software is the same as programmed in the hardware. It also ignores the interfaces that are in the STP BLK state. **Note** The **private-vlan** command does not support the **brief** or **detail** option.
show consistency-checker port-state [**module** *module-number*] [**brief** | **detail**]	Verifies the port state for the specified module.
show consistency-checker stp-state vlan *vlan-id* [**brief** | **detail**]	Determines whether the spanning tree state in the software is the same as programmed in the hardware. This command is run only on interfaces that are operational (up).

Multicast Consistency Checker

The multicast consistency checker is a single-route consistency checker for Layer 2 and Layer 3 routes for verifying the state of multicast routes. The multicast consistency checker executes the show commands in each component, parses the relevant information, and then compares the processed information against the other components to check for inconsistencies. The multicast consistency checker commands terminate upon encountering a failure. The **show consistency-checker l2 multicast group** and **show consistency-checker l3 multicast group** commands return the differences in the expected value and the actual value.

The commands support the following output formats:

- **verbose**: Displays the results in text format.
- **detail**: Displays the results in JSON format.
- **brief**: Displays the results in JSON format with minimal details.

The multicast consistency checker verifies the programming consistency of the following Layer 2 components:

- IGMP snooping

- MFDM
- MFIBPI
- MFIBPD
- Hardware tables

The multicast consistency checker verifies the programming consistency of the following Layer 3 components:

- PIM
- MRIB
- IGMP snooping
- MFDM
- MFIBPI
- MFIBPD
- Hardware tables

Output Examples for Multicast Consistency Checker Commands

The following is an example of IGMP snooping output:

```
switch# show ip igmp snooping groups 225.12.12.28 225.12.12.28 vlan 222
Type: S - Static, D - Dynamic, R - Router port, F - Fabricpath core port
Vlan  Group Address      Ver  Type  Port list
222   225.12.12.28       v3   D     Eth1/2 Eth1/3 Po12 Po100 Po18
```

The following is an example of MFDM output:

```
switch# show forwarding distribution l2 multicast vlan 222 group 225.12.12.28 source
225.12.12.28
Vlan: 222, Group: 225.12.12.28, Source: 225.12.12.28
  Outgoing Interface List Index: 4
  Reference Count: 204
  Num L3 usages: 4
  Platform Index: 0xa00004
  Vpc peer link exclude flag set
  Number of Outgoing Interfaces: 5
    Ethernet1/2
    Ethernet1/3
    port-channel12
    port-channel18
    port-channel100
```

The following is an example of comparing IGMP snooping with MFDM (passed):

```
**********************************************************************
Comparing IGMP Snooping with MFDM
**********************************************************************
L2 Eth Receivers :
IGMP Snooping: 1/2, 1/3
MFDM: 1/2, 1/3
```

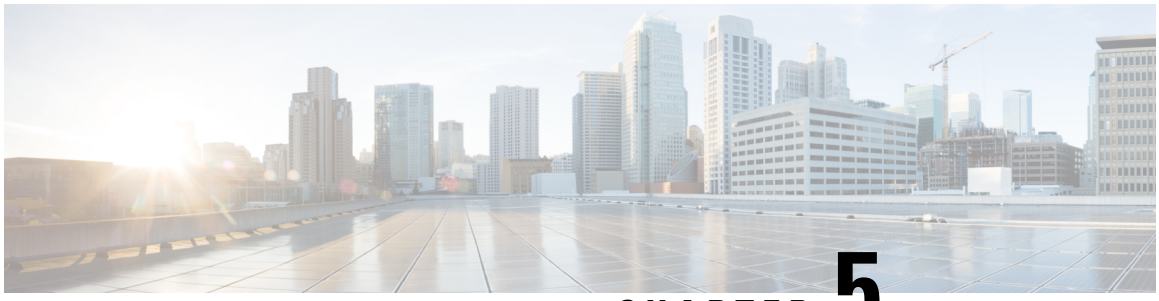
```
L2 PC Receivers :
IGMP Snooping: 100, 12, 18
MFDM: 12, 100, 18

CC between IGMP Snooping and MFDM PASSED
```

The following is an example of comparing IGMP snooping with MFDM (failed):

```
***********************************************************************
Comparing IGMP Snooping with MFDM
***********************************************************************
L2 Eth Receivers:
IGMP Snooping: 1/2, 1/3
MFDM: 1/2,  1/3

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
L2 PC Receivers:
IGMP Snooping: 100, 12, 18
MFDM: 12, 100, 16
Consistency check failed!!!
Missing elements are: 18
Additional elements are: 16
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```



CHAPTER 5

Using PowerOn Auto Provisioning

Information About PowerOn Auto Provisioning

PowerOn Auto Provisioning (POAP) automates the process of upgrading software images and installing configuration files on Cisco Nexus switches that are being deployed in the network for the first time.

When a Cisco Nexus Series switch with the POAP feature boots and does not find the startup configuration, the switch enters POAP mode and checks for a USB device containing the configuration script file. If it finds one, it checks that device to see if it also contains the software image files and the switch configuration file.

If the switch does not find a USB device, or if the USB device does not contain the needed image files or switch configuration file, the switch also locates a DHCP server and bootstraps itself with its interface IP address, gateway, and DNS server IP addresses. The switch then obtains the IP address of a TFTP server or the URL of an HTTP server from which it downloads the necessary configuration files.

Note The DHCP information is used only during the POAP process if any configuration files are unavailable on the USB device.

If the backup configuration file does not have the administrative username and the password, POAP causes a console lockout after completion. It is a mandatory step to add the username and the password in the configuration file.

Disabling POAP

POAP is enabled when there is no configuration in the system. It runs as a part of bootup. However, you can bypass POAP enablement during initial setup. If you want to disable POAP permanently (even when there is no configuration in the system), you can use the 'system no poap' command. This command ensures that POAP is not started during the next boot (even if there is no configuration). To enable POAP, use the 'system poap'

command or the 'write erase poap' command. The 'write erase poap' command erases the POAP flag and enables POAP.

- Example: Disabling POAP

```
switch# system no poap
switch# sh boot
Current Boot Variables:
 sup-1
NXOS variable = bootflash:/nxos.9.2.1.125.bin
Boot POAP Disabled

POAP permanently disabled using 'system no poap'


Boot Variables on next reload:

sup-1
NXOS variable = bootflash:/nxos.9.2.1.125.bin
Boot POAP Disabled

POAP permanently disabled using 'system no poap'


switch# sh system poap
System-wide POAP is disabled  using exec command 'system no poap'
POAP will be bypassed on write-erase reload.
(Perpetual POAP cannot be enabled when system-wide POAP is disabled)
```

- Example: Enabling POAP

```
switch# system poap

switch# sh system poap

System-wide POAP is enabled
```

- Example: Erase POAP

```
switch# write erase poap
This command will erase the system wide POAP disable flag only if it is set.
Do you wish to proceed anyway? (y/n)  [n] y
System wide POAP disable flag erased.

switch# sh system poap
System-wide POAP is enabled
```

Guidelines and Limitations for POAP

- The Cisco Nexus switch software image must support POAP for this feature to function.
- POAP can be triggered even when the startup-config is present using the **boot poap enable** command.
- If a LACP Layer 3 port-channel is configured on an uplink device connected to the Cisco Nexus device that is being bootstrapped using POAP, the port-channel is not active because all the member links are in a suspended state. Therefore, the Cisco Nexus device that is being bootstrapped using POAP cannot

reach the DHCP server or any other infrastructure device needed for POAP. To work around this issue, configure a static L3 port-channel on the uplink device connected to the Cisco Nexus device that is being bootstrapped using POAP.

- If you use POAP to bootstrap a Cisco Nexus device that is a part of a vPC pair using static port-channels on the VPC links, the Cisco Nexus device activates all of its links upon POAP startup. The dually connected device at the end of the VPC links might start sending some or all of its traffic to the port-channel member links connected to the Cisco Nexus device, and the traffic would be lost.

  To work around this issue, you can configure LACP on the vPC links so that the links do not incorrectly start forwarding traffic to the Cisco Nexus device that is being bootstrapped using POAP.

- If you use POAP to bootstrap a Cisco Nexus device that is connected downstream to a Cisco Nexus Series 7000 device through a LACP port-channel, the Cisco Nexus 7000 Series device defaults to suspend its member port if it cannot bundle it as a part of a port-channel. To work around this issue, configure the Cisco Nexus 7000 Series device to not suspend its member ports using the no lacp suspend-individual command from interface configuration mode.

- Important POAP updates are logged in the syslog and are available from the serial console.

- Critical POAP errors are logged to the bootflash. The filename format is *date-time*_poap_*PID*_[init,1,2].log, where *date-time* is in the YYYYMMDD_hhmmss format and *PID* is the process ID.

- Script logs are saved in the bootflash directory. The filename format is *date-time*_poap_*PID*_script.log, where *date-time* is in the YYYYMMDD_hhmmss format and *PID* is the process ID.

- The Scheduler configuration cannot be replayed using POAP. The reason that the Scheduler configuration cannot be replayed is that it is associated with the user (for example "admin") that was logged in when the Scheduler configuration was created. Because the configuration replay using POAP is not associated with any specific user, the scheduler configuration cannot be replayed and fails.

  Instead of configuring the Scheduler, configure the Embedded Event Manager (EEM). An EEM configuration can be downloaded and replayed using POAP.

- You can bypass password and basic POAP configuration by using the **skip** option at the POAP prompt.

  When you use the **skip** option, no password will be configured for the **admin** user. The **copy running-config startup-config** command will be blocked until a valid password is set for the **admin** user.

- The certificates (for example SSL) or configuration that are needed to be applied to the switch should be present in the configuration file.

- The syntax of the poap_script.py file should be validated using any python validation tool before using the file for POAP. Otherwise, if the poap_script.py file is edited and has a syntax error, the POAP process will exit without giving an error.

- Beginning with NX-OS 7.0(3)I7(4), RFC 3004 (User Class Option for DHCP) is supported. This enables POAP to support user-class option 77 for DHCPv4 and user-class option 15 for DHCPv6. The text displayed for the user class option for both DHCPv4 and DHCPv6 is "Cisco-POAP".

  - With RFC 3004 (User Class Option for DHCP) support, POAP over IPv6 is supported on Nexus 3000 switches.

    - Beginning with NX-OS 9.2(2), POAP over IPv6 is supported on N3K-C36180YC-R and N3K-C3636C-R switches.

The POAP over IPv6 feature enables the POAP process to use IPv6 when IPv4 fails. The feature is designed to cycle between IPv4 and IPv6 protocols when a connection failure occurs.

- For secure POAP, ensure that DHCP snooping is enabled.
- To support POAP, set firewall rules to block unintended or malicious DHCP servers.
- To maintain system security and make POAP more secure, configure the following:
  - Enable DHCP snooping.
  - Set firewall rules to block unintended or malicious DHCP servers.
- POAP is supported on both MGMT ports and in-band ports.

Setting Up the Network Environment To Use POAP

SUMMARY STEPS

1. Modify the basic configuration script provided by Cisco or create your own script. For information, see the *Python Scripting and API Configuration Guide*.
2. Every time you make a change to the configuration script, ensure that you recalculate the MD5 checksum by running **# f=poap_nexus_script.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i "s/^#md5sum=.*/#md5sum=\"$(md5sum $f.md5 | sed 's/ .*//')\"/" $f** using a bash shell. For more information, see the *Python API Reference Guide*
3. (Optional) Put the POAP configuration script and any other desired software image and switch configuration files on a USB device accessible to the switch.
4. Deploy a DHCP server and configure it with the interface, gateway, and TFTP server IP addresses and a bootfile with the path and name of the configuration script file. (This information is provided to the switch when it first boots.)
5. Deploy a TFTP or HTTP server to host the configuration script. In order to trigger the HTTP request to the server, prefix HTTP:// to the TFTP server name. HTTPS is not supported.
6. Add the URL portion into the TFTP script name to show correct path to the file name.
7. Deploy one or more servers to host the software images and configuration files.

DETAILED STEPS

Step 1 Modify the basic configuration script provided by Cisco or create your own script. For information, see the *Python Scripting and API Configuration Guide*.

Step 2 Every time you make a change to the configuration script, ensure that you recalculate the MD5 checksum by running **# f=poap_nexus_script.py ; cat $f | sed '/^#md5sum/d' > $f.md5 ; sed -i "s/^#md5sum=.*/#md5sum=\"$(md5sum $f.md5 | sed 's/ .*//')\"/" $f** using a bash shell. For more information, see the *Python API Reference Guide*

Step 3 (Optional) Put the POAP configuration script and any other desired software image and switch configuration files on a USB device accessible to the switch.

Step 4 Deploy a DHCP server and configure it with the interface, gateway, and TFTP server IP addresses and a bootfile with the path and name of the configuration script file. (This information is provided to the switch when it first boots.)

You do not need to deploy a DHCP server if all software image and switch configuration files are on the USB device.

Step 5 Deploy a TFTP or HTTP server to host the configuration script. In order to trigger the HTTP request to the server, prefix HTTP:// to the TFTP server name. HTTPS is not supported.

Step 6 Add the URL portion into the TFTP script name to show correct path to the file name.

Step 7 Deploy one or more servers to host the software images and configuration files.

Configuring a Switch Using POAP

Before you begin

Make sure that the network environment is set up to use POAP. For more information, see the Setting Up the Network Environment To Use POAP, on page 36 section immediately preceeding this section.

SUMMARY STEPS

1. Install the switch in the network.
2. Power on the switch.
3. (Optional) If you want to exit POAP mode and enter the normal interactive setup script, enter **y** (yes).
4. (Optional) If you want to bypass password and basic POAP configuration, enter **skip**.

DETAILED STEPS

Step 1 Install the switch in the network.

Step 2 Power on the switch.

If no configuration file is found, the switch boots in POAP mode and displays a prompt that asks if you want to abort POAP and continue with a normal setup.

No entry is required to continue to boot in POAP mode.

Step 3 (Optional) If you want to exit POAP mode and enter the normal interactive setup script, enter **y** (yes).

The switch boots, and the POAP process begins.

Step 4 (Optional) If you want to bypass password and basic POAP configuration, enter **skip**.

POAP is aborted and password configuration is skipped.

What to do next

Verify the configuration.

Verifying the Device Configuration

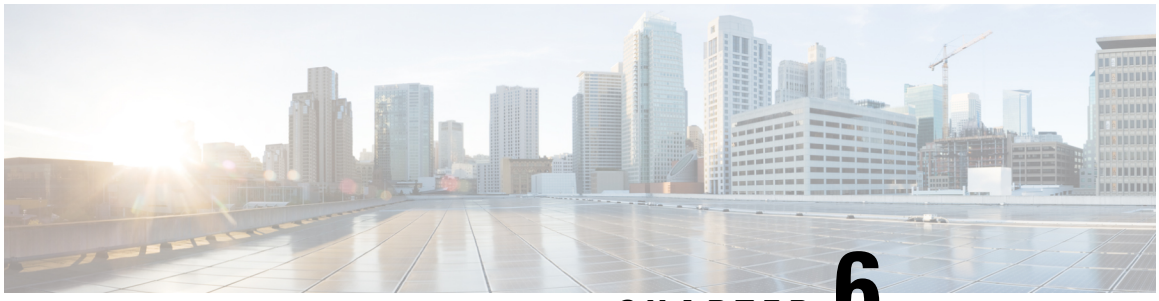
To verify the configuration after bootstrapping the device using POAP, use one of the following commands:

Command	Purpose
show running-config	Displays the running configuration.
show startup-config	Displays the startup configuration.
show time-stamp running-config last-changed	Displays the timestamp when the running configuration was last changed.

For detailed information about the fields in the output from these commands, see the Cisco Nexus command reference for your device.

Related Documents for POAP

Related Topic	Document Title
Configuration Script	*Cisco Nexus 3000 Series NX-OS Python API Reference Guide*
DHCP Options and BOOTP Vendor Extensions	RFC2132—http://tools.ietf.org/html/rfc2132
TFTP Server Address Option for DHCPv4	RFC5859—http://tools.ietf.org/html/rfc5859



CHAPTER 6

Using the Device File Systems, Directories, and Files

Information About the Device File Systems, Directories, and Files

This section describes file systems, directories, and files on the Cisco NX-OS device.

File Systems

The syntax for specifying a local file system is *filesystem*:[//*modules*/].
This table describes file systems that you can reference on your device.

Table 10: File System Syntax Components

File System Name	Module	Description
bootflash	sup-active sup-local	Internal CompactFlash memory located on the active supervisor module used for storing image files, configuration files, and other miscellaneous files. The initial default directory is bootflash.

File System Name	Module	Description
volatile	—	Volatile random-access memory (VRAM) located on a supervisor module used for temporary or pending changes.
log	—	Memory on the active supervisor that stores logging file statistics.
system	—	Memory on a supervisor module used for storing the running-configuration file.
debug	—	Memory on a supervisor module used for debug logs.

Directories

You can create directories on bootflash: and external flash memory (slot0:, usb1:, and usb2:). You can navigate through these directories and use them for files.

Files

You create and access files on bootflash:. volatile:, slot0:, usb1:, and usb2: file systems. You can only access files on the system: file systems. You can use the debug: file system for debug log files specified in the **debug logfile** command.

You can download files, such as system image files, from remote servers using FTP, Secure Copy (SCP), Secure Shell FTP (SFTP), and TFTP. You can also copy files from an external server to the device, because the device can act as an SCP server.

Guidelines and Limitations

Guidelines and limitations for device file systems, directories, and files are as follows:

- The **show tech-support details** command cannot be terminated using Ctrl+Z. Instead, use Ctrl+C to terminate the command.

Working with Directories

This section describes how to work with directories on the Cisco NX-OS device.

Identifying the Current Directory

You can display the directory name of your current directory.

SUMMARY STEPS

1. **pwd**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**pwd** **Example:** `switch# pwd`	Displays the name of your current directory.

Changing the Current Directory

You can change the current directory for file system operations. The initial default directory is bootflash:.

SUMMARY STEPS

1. (Optional) **pwd**
2. **cd** {*directory* | *filesystem*:[//*module*/][*directory*]}

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **pwd** **Example:** `switch# pwd`	Displays the name of your current default directory.
Step 2	**cd** {*directory* | *filesystem*:[//*module*/][*directory*]} **Example:** `switch# cd slot0:`	Changes to a new current directory. The file system, module, and directory names are case sensitive.

Creating a Directory

You can create directories in the bootflash: and flash device file systems.

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **cd** {*directory* | *filesystem*:[//*module*/][*directory*]}
3. **mkdir** [*filesystem*:[//*module*/]]*directory*

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **pwd** **Example:** `switch# pwd`	Displays the name of your current default directory.
Step 2	(Optional) **cd** {*directory* | *filesystem*:[//*module*/][*directory*]} **Example:** `switch# cd slot0:`	Changes to a new current directory. The file system, module, and directory names are case sensitive.
Step 3	**mkdir** [*filesystem*:[//*module*/]]*directory* **Example:** `switch# mkdir test`	Creates a new directory. The *filesystem* argument is case sensitive. The *directory* argument is alphanumeric, case sensitive, and has a maximum of 64 characters.

Displaying Directory Contents

You can display the contents of a directory.

SUMMARY STEPS

1. **dir** [*directory* | *filesystem*:[//*module*/][*directory*]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**dir** [*directory* | *filesystem*:[//*module*/][*directory*]] **Example:** `switch# dir bootflash:test`	Displays the directory contents. The default is the current working directory. The file system and directory names are case sensitive.

Deleting a Directory

You can remove directories from the file systems on your device.

Before you begin

Ensure that the directory is empty before you try to delete it.

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **dir** [*filesystem* :[//*module*/][*directory*]]
3. **rmdir** [*filesystem* :[//*module*/]]*directory*

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **pwd** **Example:** `switch# pwd`	Displays the name of your current default directory.
Step 2	(Optional) **dir** [*filesystem* **:**[*//module/*][*directory*]] **Example:** `switch# dir bootflash:test`	Displays the contents of the current directory. The file system, module, and directory names are case sensitive. If the directory is not empty, you must delete all the files before you can delete the directory.
Step 3	**rmdir** [*filesystem* **:**[*//module/*]]*directory* **Example:** `switch# rmdir test`	Deletes a directory. The file system and directory name are case sensitive.

Accessing Directories on Standby Supervisor Modules

This example shows how to list the files on the standby supervisor module:

```
switch# dir bootflash://sup-remote
   12198912     Aug 27 16:29:18 2003  m9500-sf1ek9-kickstart-mzg.1.3.0.39a.bin
    1864931     Apr 29 12:41:59 2003  dplug2
      12288     Apr 18 20:23:11 2003  lost+found/
   12097024     Nov 21 16:34:18 2003  m9500-sf1ek9-kickstart-mz.1.3.1.1.bin
   41574014     Nov 21 16:34:47 2003  m9500-sf1ek9-mz.1.3.1.1.bin

Usage for bootflash://sup-remote
   67747169 bytes used
  116812447 bytes free
  184559616 bytes total
```

This example shows how to delete a file on the standby supervisor module:

```
switch# delete bootflash://sup-remote/aOldConfig.txt
```

Working with Files

This section describes how to work with files on the Cisco NX-OS device.

Moving Files

You can move a file from one directory to another directory.

Caution If a file with the same name already exists in the destination directory, that file is overwritten by the moved file.

You can use the **move** command to rename a file by moving the file within the same directory.

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **dir** [*filesystem*:[//*module*/][*directory*]]
3. **move** [*filesystem*:[//*module*/][*directory* /] | *directory*/]*source-filename* {{*filesystem*:[//*module*/][*directory* /] | *directory*/}[*target-filename*] | *target-filename*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **pwd** **Example:** `switch# pwd`	Displays the name of your current default directory.
Step 2	(Optional) **dir** [*filesystem*:[//*module*/][*directory*]] **Example:** `switch# dir bootflash`	Displays the contents of the current directory. The file system and directory name are case sensitive.
Step 3	**move** [*filesystem*:[//*module*/][*directory* /]	*directory*/]*source-filename* {{*filesystem*:[//*module*/][*directory* /]	*directory*/}[*target-filename*]	*target-filename*} **Example:** `switch# move test old_tests/test1`	Moves a file. The file system, module, and directory names are case sensitive. The *target-filename* argument is alphanumeric, case sensitive, and has a maximum of 64 characters. If the *target-filename* argument is not specified, the filename defaults to the *source-filename* argument value.

Copying Files

This example shows how to copy the file called samplefile from the root directory of the slot0: file system to the mystorage directory:

```
switch# copy slot0:samplefile slot0:mystorage/samplefile
```

This example shows how to copy a file from the current directory level:

```
switch# copy samplefile mystorage/samplefile
```

This example shows how to copy a file from the active supervisor module bootflash to the standby supervisor module bootflash:

```
switch# copy bootflash:system_image bootflash://sup-2/system_image
```

You can also use the **copy** command to upload and download files from the slot0: or bootflash: file system to or from a FTP, TFTP, SFTP, or SCP server.

Copying Files to the HTTP Server

You can copy a file from boot flash, running configuration, or startup configuration to the HTTP server in the default path of server using the new feature HTTP PUT. The HTTP PUT functionality copies a file into the HTTP server via the default or the management VRF. It uses the HTTP POST method to upload the files and HTTP upload can be done via v4/v6 interface to v4/v6 HTTP server.

SUMMARY STEPS

1. (Optional) **pwd**
2. **copy bootflash :**///<*filename*> **http:**// <*httpserver-ip*> /*path source-interface* <*interface*>
3. **copy bootflash :**///<*filename*> **http:**// <*httpserver-ip*>/*path vrf* <*default/management*>
4. **copy running-config http:**//<*httpserver-ip*>/*path vrf* <*default/management*>
5. **copy startup-config http:**//<*httpserver-ip*>/*path vrf* <*default/management*>

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **pwd** **Example:** `switch# pwd`	Displays the name of your current default directory.
Step 2	Required: **copy bootflash :**///<*filename*> **http:**// <*httpserver-ip*> /*path source-interface* <*interface*> **Example:** `switch# copy bootflash:///<filename> http://httpserver-ip/path source-interface <int>`	Copies the file from the boot flash into the HTTP server via the source interface.
Step 3	Required: **copy bootflash :**///<*filename*> **http:**// <*httpserver-ip*>/*path vrf* <*default/management*> **Example:** `switch# copy bootflash:///<filename> http://httpserver-ip/path vrf <default/management>`	Copies the file from the boot flash into the HTTP server via the default or the management VRF interface.
Step 4	Required: **copy running-config http:**//<*httpserver-ip*>/*path vrf* <*default/management*> **Example:** `switch# copy running-config http://httpserver-ip/path vrf <default/management>`	Copies the running configuration file into the HTTP server via the default or the management VRF interface.
Step 5	Required: **copy startup-config http:**//<*httpserver-ip*>/*path vrf* <*default/management*> **Example:** `switch# copy startup-config http://httpserver-ip/path vrf <default/management>`	Copies the startup configuration file into the HTTP server via the default or the management VRF interface.

Example

This example shows how to copy a file to an HTTP server via the default VRF interface:

```
switch# copy n3000-uk9-kickstart.6.0.2.U5.0.995.bin http://12.1.2.10/httproot vrf default
Enter username: test
Enter host password for user 'test':
301 - Moved permanently to <a href="/httproot/">/httproot/</a>Copy
complete, now saving to disk (please wait)...
switch#
```

Deleting Files

You can delete a file from a directory.

SUMMARY STEPS

1. (Optional) **dir** [*filesystem*:[//*module*/][*directory*]]
2. **delete** {*filesystem*:[//*module*/][*directory*/] | *directory*/}*filename*

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **dir** [*filesystem*:[//*module*/][*directory*]] **Example:** `switch# dir bootflash`	Displays the contents of the current directory. The file system and directory name are case sensitive.
Step 2	**delete** {*filesystem*:[//*module*/][*directory*/]	*directory*/}*filename* **Example:** `switch# delete test old_tests/test1`	Deletes a file. The file system, module, and directory names are case sensitive. The *source-filename* argument is case sensitive. **Caution** If you specify a directory, the **delete** command deletes the entire directory and all its contents.

Displaying File Contents

You can display the contents of a file.

SUMMARY STEPS

1. **show file** [*filesystem*:[//*module*/]][*directory*/]*filename*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show file** [*filesystem*:[//*module*/]][*directory*/]*filename* **Example:**	Displays the file contents.

	Command or Action	Purpose
	`switch# show file bootflash:test-results`	

Displaying File Checksums

You can display checksums to check the file integrity.

SUMMARY STEPS

1. **show file** [*filesystem*:[//*module*/]][*directory*/]*filename* {**cksum** | **md5sum**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show file** [*filesystem*:[//*module*/]][*directory*/]*filename* {**cksum**	**md5sum**} **Example:** `switch# show file bootflash:trunks2.cfg cksum`	Displays the checksum or MD5 checksum of the file.

Compressing and Uncompressing Files

You can compress and uncompress files on your Cisco NX-OS device using Lempel-Ziv 1977 (LZ77) coding.

SUMMARY STEPS

1. (Optional) **dir** [*filesystem*:[//*module*/]*directory*]]
2. **gzip** [*filesystem*:[//*module*/][*directory*/] | *directory*/]*filename*
3. **gunzip** [*filesystem*:[//*module*/][*directory*/] | *directory*/]*filename* **.gz**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **dir** [*filesystem*:[//*module*/]*directory*]] **Example:** `switch# dir bootflash:`	Displays the contents of the current directory. The file system and directory name are case sensitive.
Step 2	**gzip** [*filesystem*:[//*module*/][*directory*/]	*directory*/]*filename* **Example:** `switch# gzip show_tech`	Compresses a file. After the file is compressed, it has a .gz suffix.
Step 3	**gunzip** [*filesystem*:[//*module*/][*directory*/]	*directory*/]*filename* **.gz** **Example:** `switch# gunzip show_tech.gz`	Uncompresses a file. The file to uncompress must have the .gz suffix. After the file is uncompressed, it does not have the .gz suffix.

Displaying the Last Lines in a File

You can display the last lines of a file.

SUMMARY STEPS

1. **tail** [*filesystem*:[//*module*/]][*directory*/]*filename* [*lines*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**tail** [*filesystem*:[//*module*/]][*directory*/]*filename* [*lines*] **Example:** `switch# tail ospf-gr.conf`	Displays the last lines of a file. The default number of lines is 10. The range is from 0 to 80 lines.

Redirecting show Command Output

This example shows how to direct the output to a file on the bootflash: file system:

```
switch# show interface > bootflash:switch1-intf.cfg
```

This example shows how to direct the output to a file on external flash memory:

```
switch# show interface > slot0:switch-intf.cfg
```

This example shows how to direct the output to a file on a TFTP server:

```
switch# show interface > tftp://10.10.1.1/home/configs/switch-intf.cfg
Preparing to copy...done
```

This example shows how to direct the output of the **show tech-support** command to a file:

```
switch# show tech-support > Samplefile
Building Configuration ...
switch# dir
    1525859     Jul 04 00:51:03 2003 Samplefile
Usage for volatile://
    1527808 bytes used
   19443712 bytes free
   20971520 bytes total
```

Finding Files

This example shows how to find a file in the current default directory:

```
switch# find smm_shm.cfg
/usr/bin/find: ./lost+found: Permission denied
./smm_shm.cfg
```

```
./newer-fs/isan/etc/routing-sw/smm_shm.cfg
./newer-fs/isan/etc/smm_shm.cfg
```

Working with Archive Files

The Cisco NX-OS software supports archive files. You can create an archive file, append files to an existing archive file, extract files from an archive file, and list the files in an archive file.

Creating an Archive Files

You can create an archive file and add files to it. You can specify the following compression types:

- bzip2
- gzip
- Uncompressed

The default is gzip.

SUMMARY STEPS

1. **tar create** {**bootflash:** | **volatile:**}*archive-filename* [**absolute**] [**bz2-compress**] [**gz-compress**] [**remove**] [**uncompressed**] [**verbose**] *filename-list*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**tar create** {**bootflash:**	**volatile:**}*archive-filename* [**absolute**] [**bz2-compress**] [**gz-compress**] [**remove**] [**uncompressed**] [**verbose**] *filename-list*	Creates an archive file and adds files to it. The filename is alphanumeric, not case sensitive, and has a maximum length of 240 characters. The **absolute** keyword specifies that the leading backslash characters (\) should not be removed from the names of the files added to the archive file. By default, the leading backslash characters are removed. The **bz2-compress**, **gz-compress**, and **uncompressed** keywords determine the compression utility used when files are added, or later appended, to the archive and the decompression utility to use when extracting the files. If you do not specify an extension for the archive file, the defaults are as follows: • For **bz2-compress**, the extension is .tar.bz2. • For **gz-compress**, the extension is .tar.gz. • For **uncompressed**, the extension is .tar. The **remove** keyword specifies that the Cisco NX-OS software should delete the files from the file system after

	Command or Action	Purpose
		adding them to the archive. By default, the files are not deleted. The **verbose** keyword specifies that the Cisco NX-OS software should list the files as they are added to the archive. By default, the files are listed as they are added.

Example

This example shows how to create a gzip compressed archive file:

```
switch# tar create bootflash:config-archive gz-compress bootflash:config-file
```

Appending Files to an Archive File

You can append files to an existing archive file on your Cisco NX-OS device.

Before you begin

You have created an archive file on your Cisco NX-OS device.

SUMMARY STEPS

1. **tar append** {**bootflash:** | **volatile:**}*archive-filename* [**absolute**] [**remove**] [**verbose**] *filename-list*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**tar append** {**bootflash:**	**volatile:**}*archive-filename* [**absolute**] [**remove**] [**verbose**] *filename-list*	Adds files to an existing archive file. The archive filename is not case sensitive. The **absolute** keyword specifies that the leading backslash characters (\) should not be removed from the names of the files added to the archive file. By default, the leading backslash characters are removed. The **remove** keyword specifies that the Cisco NX-OS software should delete the files from the filesystem after adding them to the archive. By default, the files are not deleted. The **verbose** keyword specifies that the Cisco NX-OS software should list the files as they are added to the archive. By default, the files are listed as they are added.

Example

This example shows how to append a file to an existing archive file:

```
switch# tar append bootflash:config-archive.tar.gz bootflash:new-config
```

Extracting Files from an Archive File

You can extract files to an existing archive file on your Cisco NX-OS device.

Before you begin

You have created an archive file on your Cisco NX-OS device.

SUMMARY STEPS

1. **tar extract** {**bootflash:** | **volatile:**}*archive-filename* [**keep-old**] [**screen**] [**to** {**bootflash:** | **volatile:**}[*/directory-name*]] [**verbose**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**tar extract** {**bootflash:** | **volatile:**}*archive-filename* [**keep-old**] [**screen**] [**to** {**bootflash:** | **volatile:**}[*/directory-name*]] [**verbose**]	Extracts files from an existing archive file. The archive filename is not case sensitive. The **keep-old** keyword indicates that the Cisco NX-OS software should not overwrite files with the same name as the files being extracted. The **screen** keyword specifies that the Cisco NX-OS software should display the contents of the extracted files to the terminal screen. The **to** keyword specifies the target file system. You can include a directory name. The directory name is alphanumeric, case sensitive, and has a maximum length of 240 characters. The **verbose** keyword specifies that the Cisco NX-OS software should display the names of the files as they are extracted.

Example

This example shows how to extract files from an existing archive file:

```
switch# tar extract bootflash:config-archive.tar.gz
```

Displaying the Filenames in an Archive File

You can display the names of the files in an archive files using the **tar list** command.

tar list {**bootflash:** | **volatile:**}*archive-filename*

The archive filename is not case sensitive.

```
switch# tar list bootflash:config-archive.tar.gz
config-file
new-config
```

Examples of Using the File System

This section includes example of using the file system on the Cisco NX-OS device.

Accessing Directories on Standby Supervisor Modules

This example shows how to list the files on the standby supervisor module:

```
switch# dir bootflash://sup-remote
   12198912     Aug 27 16:29:18 2003  m9500-sf1ek9-kickstart-mzg.1.3.0.39a.bin
    1864931     Apr 29 12:41:59 2003  dplug2
      12288     Apr 18 20:23:11 2003  lost+found/
   12097024     Nov 21 16:34:18 2003  m9500-sf1ek9-kickstart-mz.1.3.1.1.bin
   41574014     Nov 21 16:34:47 2003  m9500-sf1ek9-mz.1.3.1.1.bin

Usage for bootflash://sup-remote
   67747169 bytes used
  116812447 bytes free
  184559616 bytes total
```

This example shows how to delete a file on the standby supervisor module:

```
switch# delete bootflash://sup-remote/aOldConfig.txt
```

Moving Files

You can move a file from one directory to another directory.

Caution If a file with the same name already exists in the destination directory, that file is overwritten by the moved file.

You can use the **move** command to rename a file by moving the file within the same directory.

SUMMARY STEPS

1. (Optional) **pwd**
2. (Optional) **dir** [*filesystem*:[//*module*/][*directory*]]
3. **move** [*filesystem*:[//*module*/][*directory* /] | *directory*/]*source-filename* {{*filesystem*:[//*module*/][*directory* /] | *directory*/}[*target-filename*] | *target-filename*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **pwd** **Example:** `switch# pwd`	Displays the name of your current default directory.
Step 2	(Optional) **dir** [*filesystem*:[//*module*/][*directory*]] **Example:** `switch# dir bootflash`	Displays the contents of the current directory. The file system and directory name are case sensitive.
Step 3	**move** [*filesystem*:[//*module*/][*directory* /] | *directory*/]*source-filename* {{*filesystem*:[//*module*/][*directory* /] | *directory*/}[*target-filename*] | *target-filename*} **Example:** `switch# move test old_tests/test1`	Moves a file. The file system, module, and directory names are case sensitive. The *target-filename* argument is alphanumeric, case sensitive, and has a maximum of 64 characters. If the *target-filename* argument is not specified, the filename defaults to the *source-filename* argument value.

Copying Files

This example shows how to copy the file called samplefile from the root directory of the slot0: file system to the mystorage directory:

```
switch# copy slot0:samplefile slot0:mystorage/samplefile
```

This example shows how to copy a file from the current directory level:

```
switch# copy samplefile mystorage/samplefile
```

This example shows how to copy a file from the active supervisor module bootflash to the standby supervisor module bootflash:

```
switch# copy bootflash:system_image bootflash://sup-2/system_image
```

You can also use the **copy** command to upload and download files from the slot0: or bootflash: file system to or from a FTP, TFTP, SFTP, or SCP server.

Deleting a Directory

You can remove directories from the file systems on your device.

Before you begin

Ensure that the directory is empty before you try to delete it.

SUMMARY STEPS

1. (Optional) **pwd**

2. (Optional) **dir** [*filesystem* **:**[*//module/*][*directory*]]
3. **rmdir** [*filesystem* **:**[*//module/*]]*directory*

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **pwd** **Example:** `switch# pwd`	Displays the name of your current default directory.
Step 2	(Optional) **dir** [*filesystem* **:**[*//module/*][*directory*]] **Example:** `switch# dir bootflash:test`	Displays the contents of the current directory. The file system, module, and directory names are case sensitive. If the directory is not empty, you must delete all the files before you can delete the directory.
Step 3	**rmdir** [*filesystem* **:**[*//module/*]]*directory* **Example:** `switch# rmdir test`	Deletes a directory. The file system and directory name are case sensitive.

Displaying File Contents

This example shows how to display the contents of a file on an external flash device:

```
switch# show file slot0:test
configure terminal
interface ethernet 1/1
no shutdown
end
show interface ethernet 1/1
```

This example shows how to display the contents of a file that resides in the current directory:

```
switch# show file myfile
```

Displaying File Checksums

This example shows how to display the checksum of a file:

```
switch# show file bootflash:trunks2.cfg cksum
583547619
```

This example shows how to display the MD5 checksum of a file:

```
switch# show file bootflash:trunks2.cfg md5sum
3b94707198aabefcf46459de10c9281c
```

Compressing and Uncompressing Files

This example shows how to compress a file:

```
switch# dir
    1525859     Jul 04 00:51:03 2003 Samplefile
...
switch# gzip volatile:Samplefile
switch# dir
     266069     Jul 04 00:51:03 2003 Samplefile.gz
...
```

This example shows how to uncompress a compressed file:

```
switch# dir
     266069     Jul 04 00:51:03 2003 Samplefile.gz
...
switch# gunzip samplefile
switch# dir
    1525859     Jul 04 00:51:03 2003 Samplefile
...
```

Redirecting show Command Output

This example shows how to direct the output to a file on the bootflash: file system:

```
switch# show interface > bootflash:switch1-intf.cfg
```

This example shows how to direct the output to a file on external flash memory:

```
switch# show interface > slot0:switch-intf.cfg
```

This example shows how to direct the output to a file on a TFTP server:

```
switch# show interface > tftp://10.10.1.1/home/configs/switch-intf.cfg
Preparing to copy...done
```

This example shows how to direct the output of the **show tech-support** command to a file:

```
switch# show tech-support > Samplefile
Building Configuration ...
switch# dir
    1525859     Jul 04 00:51:03 2003 Samplefile
Usage for volatile://
    1527808 bytes used
   19443712 bytes free
   20971520 bytes total
```

Finding Files

This example shows how to find a file in the current default directory:

```
switch# find smm_shm.cfg
/usr/bin/find: ./lost+found: Permission denied
./smm_shm.cfg
./newer-fs/isan/etc/routing-sw/smm_shm.cfg
./newer-fs/isan/etc/smm_shm.cfg
```

Default Settings for File System Parameters

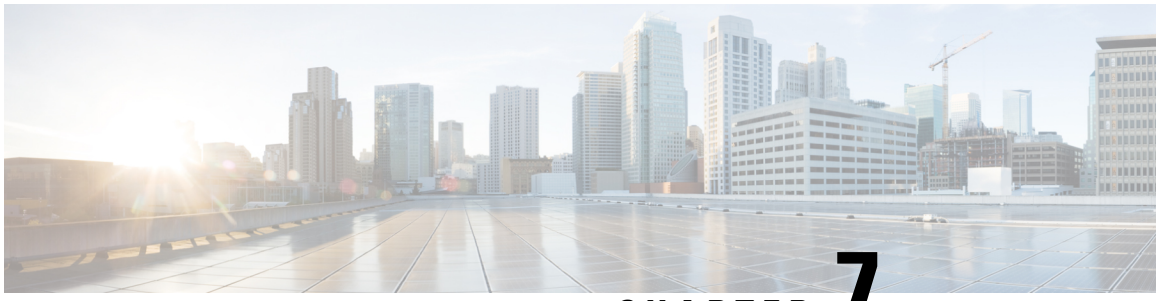
This table lists the default settings for the file system parameters.

Table 11: Default File System Settings

Parameters	Default
Default filesystem	bootflash:

Additional References for File Systems

This section includes additional information related to the file systems.



CHAPTER 7

Working with Configuration Files

Information About Configuration Files

Configuration files contain the Cisco NX-OS software commands used to configure the features on a Cisco NX-OS device. Commands are parsed (translated and executed) by the Cisco NX-OS software when the system is booted (from the startup-config file) or when you enter commands at the CLI in a configuration mode.

To change the startup configuration file, you can either save the running-configuration file to the startup configuration using the **copy running-config startup-config** command or copy a configuration file from a file server to the startup configuration.

Types of Configuration Files

The Cisco NX-OS software has two types of configuration files, running configuration and startup configuration. The device uses the startup configuration (startup-config) during device startup to configure the software features. The running configuration (running-config) contains the current changes that you make to the startup-configuration file. The two configuration files can be different. You might want to change the device configuration for a short time period rather than permanently. In this case, you would change the running configuration by using commands in global configuration mode but not save the changes to the startup configuration.

To change the running configuration, use the **configure terminal** command to enter global configuration mode. As you use the Cisco NX-OS configuration modes, commands generally are executed immediately and are saved to the running configuration file either immediately after you enter them or when you exit a configuration mode.

To change the startup-configuration file, you can either save the running configuration file to the startup configuration or download a configuration file from a file server to the startup configuration.

Related Topics

Saving the Running Configuration to the Startup Configuration, on page 58
Downloading the Startup Configuration From a Remote Server, on page 61

Managing Configuration Files

This section describes how to manage configuration files.

Copying Configuration Files to the Startup Configuration

You can directly copy configuration files, through FTP or SCP, to the startup configuration without reloading the device.

SUMMARY STEPS

1. **copy** *scheme://[user@]server/[url/]filename* **startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**copy** *scheme://[user@]server/[url/]filename* **startup-config** **Example:** `switch(boot)# copy scp://user@123.40.56.78/wp/user/abc_cfg startup-config`	Copies the configuration file directly through SCP or FTP to the startup configuration. For the *scheme* argument, you can enter either **ftp** or **scp**. The *user@* argument is your username, the *server* argument is the address or name of the remote server, and the *url* argument is the path to the source file on the remote server. The *user@*, *server*, *url*, and *filename* arguments are case sensitive. This process does not require you to reload the device.

Saving the Running Configuration to the Startup Configuration

You can save the running configuration to the startup configuration to save your changes for the next time you that reload the device.

SUMMARY STEPS

1. (Optional) **show running-config**
2. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **show running-config** **Example:**	Displays the running configuration.

	Command or Action	Purpose
	`switch# show running-config`	
Step 2	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Copying a Configuration File to a Remote Server

You can copy a configuration file stored in the internal memory to a remote server as a backup or to use for configuring other Cisco NX-OS devices.

SUMMARY STEPS

1. **copy running-config** *scheme*:*//server/*[*url /*]*filename*
2. **copy startup-config** *scheme*:*//server/*[*url /*]*filename*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**copy running-config** *scheme*:*//server/*[*url /*]*filename* **Example:** `switch# copy running-config tftp://10.10.1.1/sw1-run-config.bak`	Copies the running-configuration file to a remote server. For the *scheme* argument, you can enter **tftp:**, **ftp:**, **scp:**, **sftp:**, **http:**, or **https:**. The *server* argument is the address or name of the remote server, and the *url* argument is the path to the source file on the remote server. The *server*, *url*, and *filename* arguments are case sensitive.
Step 2	**copy startup-config** *scheme*:*//server/*[*url /*]*filename* **Example:** `switch# copy startup-config tftp://10.10.1.1/sw1-start-config.bak`	Copies the startup-configuration file to a remote server. For the *scheme* argument, you can enter **tftp:**, **ftp:**, **scp:**, **sftp:**, **http:**, or **https:**. The *server* argument is the address or name of the remote server, and the *url* argument is the path to the source file on the remote server. The *server*, *url*, and *filename* arguments are case sensitive.

Example

This example shows how to copy the configuration file to a remote server:

```
switch# copy running-config
tftp://10.10.1.1/sw1-run-config.bak
switch# copy startup-config
tftp://10.10.1.1/sw1-start-config.bak
```

Downloading the Running Configuration From a Remote Server

You can configure your Cisco NX-OS device by using configuration files that you created on another Cisco NX-OS device and uploaded to a remote server. You then download the file from the remote server to your device using TFTP, FTP, Secure Copy (SCP), or Secure Shell FTP (SFTP) to the running configuration.

Before you begin

Ensure that the configuration file that you want to download is in the correct directory on the remote server.

Ensure that the permissions on the file are set correctly. Permissions on the file should be set to world-read.

Ensure that your Cisco NX-OS device has a route to the remote server. The Cisco NX-OS device and the remote server must be in the same subnetwork if you do not have a router or a default gateway to route traffic between subnets.

Check connectivity to the remote server using the **ping** or **ping6** command.

SUMMARY STEPS

1. **copy** *scheme***://***server*/[*url*/]*filename* **running-config**
2. (Optional) **show running-config**
3. (Optional) **copy running-config startup-config**
4. (Optional) **show startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**copy** *scheme***://***server*/[*url*/]*filename* **running-config** **Example:** `switch# copy tftp://10.10.1.1/my-config running-config`	Downloads the running-configuration file from a remote server. For the *scheme* argument, you can enter **tftp:**, **ftp:**, **scp:**, **sftp:**, **http:**, or **https:**. The *server* argument is the address or name of the remote server, and the *url* argument is the path to the source file on the remote server. The *server*, *url*, and *filename* arguments are case sensitive.
Step 2	(Optional) **show running-config** **Example:** `switch# show running-config`	Displays the running configuration.
Step 3	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.
Step 4	(Optional) **show startup-config** **Example:** `switch# show startup-config`	Displays the startup configuration.

Related Topics

Copying Files

Downloading the Startup Configuration From a Remote Server

You can configure your Cisco NX-OS device by using configuration files that you created on another Cisco NX-OS device and uploaded to a remote server. You then download the file from the remote server to your device using TFTP, FTP, Secure Copy (SCP), Secure Shell FTP (SFTP), HTTP, or HTTPS to the startup configuration.

Caution This procedure disrupts all traffic on the Cisco NX-OS device.

Before you begin

Log in to a session on the console port.

Ensure that the configuration file that you want to download is in the correct directory on the remote server.

Ensure that the permissions on the file are set correctly. Permissions on the file should be set to world-read.

Ensure that your Cisco NX-OS device has a route to the remote server. The Cisco NX-OS device and the remote server must be in the same subnetwork if you do not have a router or a default gateway to route traffic between subnets.

Check connectivity to the remote server using the **ping** or **ping6** command.

SUMMARY STEPS

1. **write erase**
2. **reload**
3. **copy** *scheme*:*//server/*[*url /*]*filename* **running-config**
4. **copy running-config startup-config**
5. (Optional) **show startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**write erase** **Example:** `switch# write erase`	Erases the startup configuration file.

Step 2	**reload**
	Example:
	`switch# reload`
	`This command will reboot the system. (y/n)?  [n]`
	`y`
	`...`
	`Enter the password for "admin": <password>`
	`Confirm the password for "admin": <password>`
	`...`
		Reloads the Cisco NX-OS device.
		Note Do not use the setup utility to configure the device.

	Command or Action	Purpose
	`Would you like to enter the basic configuration dialog (yes/no): n` `switch#`	
Step 3	**copy** *scheme*:*//server/*[*url /*]*filename* **running-config** **Example:** `switch# copy tftp://10.10.1.1/my-config running-config`	Downloads the running configuration file from a remote server. For the *scheme* argument, you can enter **tftp:**, **ftp:**, **scp:**, **sftp:**, **http:**, or **https:**. The *server* argument is the address or name of the remote server, and the *url* argument is the path to the source file on the remote server. The *server*, *url*, and *filename* arguments are case sensitive.
Step 4	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Saves the running configuration file to the startup configuration file. **Note** You can use the **copy** {**ftp:** | **scp:** | **sftp:** | **tftp:** | **http:** | **https:**}*source-url* **startup-config** command to copy a configuration file from a network server to the switch startup configuration. This command replaces the startup configuration file with the copied configuration file. Beginning with Cisco NX-OS Release 6.0(2)U2(1), the startup configuration file is stored as an ASCII text file and all commands in the configuration file are run during the next bootup to generate the binary configuration file. This is equivalent to booting with **write erase** and applying configuration commands individually on reload. Because all commands in the startup configuration file are run as configuration commands, this can delay the ASCII configuration file from taking effect.
Step 5	(Optional) **show startup-config** **Example:** `switch# show startup-config`	Displays the running configuration.

Related Topics

Copying Files

Copying Configuration Files to an External Flash Memory Device

You can copy configuration files to an external flash memory device as a backup for later use.

Before you begin

Insert the external Flash memory device into the active supervisor module.

SUMMARY STEPS

1. (Optional) **dir** {**slot0:** | **usb1:** | **usb2:**}[*directory/*]
2. **copy running-config** {**slot0:** | **usb1:** | **usb2:**}[*directory/*]*filename*
3. **copy startup-config** {**slot0:** | **usb1:** | **usb2:**}[*directory/*]*filename*

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **dir** {**slot0:** | **usb1:** | **usb2:**}[*directory/*] **Example:** `switch# dir slot0:`	Displays the files on the external flash memory device.
Step 2	**copy running-config** {**slot0:** | **usb1:** | **usb2:**}[*directory/*]*filename* **Example:** `switch# copy running-config slot0:dsn-running-config.cfg`	Copies the running configuration to an external flash memory device. The *filename* argument is case sensitive.
Step 3	**copy startup-config** {**slot0:** | **usb1:** | **usb2:**}[*directory/*]*filename* **Example:** `switch# copy startup-config slot0:dsn-startup-config.cfg`	Copies the startup configuration to an external flash memory device. The *filename* argument is case sensitive.

Related Topics

Copying Files

Copying the Running Configuration from an External Flash Memory Device

You can configure your Cisco NX-OS device by copying configuration files created on another Cisco NX-OS device and saved to an external flash memory device.

Before you begin

Insert the external flash memory device into the active supervisor module.

SUMMARY STEPS

1. (Optional) **dir** {**slot0:** | **usb1:** | **usb2:**}[*directory/*]
2. **copy** {**slot0:** | **usb1:** | **usb2:**}[*directory/*]*filename* **running-config**
3. (Optional) **show running-config**
4. (Optional) **copy running-config startup-config**
5. (Optional) **show startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **dir** {**slot0:** | **usb1:** | **usb2:**}[*directory/*] **Example:** `switch# dir slot0:`	Displays the files on the external flash memory device.
Step 2	**copy** {**slot0:** | **usb1:** | **usb2:**}[*directory/*]*filename* **running-config** **Example:** `switch# copy slot0:dsn-config.cfg running-config`	Copies the running configuration from an external flash memory device. The *filename* argument is case sensitive.
Step 3	(Optional) **show running-config** **Example:** `switch# show running-config`	Displays the running configuration.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.
Step 5	(Optional) **show startup-config** **Example:** `switch# show startup-config`	Displays the startup configuration.

Related Topics

Copying Files

Copying Configuration Files to an Internal File System

You can copy configuration files to the internal memory as a backup for later use.

SUMMARY STEPS

1. **copy running-config** [*filesystem***:**][*directory/*] | [*directory/*]*filename*
2. **copy startup-config** [*filesystem***:**][*directory/*] | [*directory/*]*filename*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**copy running-config** [*filesystem***:**][*directory/*] | [*directory/*]*filename* **Example:** `switch# copy running-config bootflash:sw1-run-config.bak`	Copies the running-configuration file to internal memory. The *filesystem*, *directory*, and *filename* arguments are case sensitive.
Step 2	**copy startup-config** [*filesystem***:**][*directory/*] | [*directory/*]*filename*	Copies the startup-configuration file to internal memory.

	Command or Action	Purpose
	Example: `switch# copy startup-config bootflash:sw1-start-config.bak`	The *filesystem*, *directory*, and *filename* arguments are case sensitive.

Related Topics

Copying Files

Rolling Back to a Previous Configuration

Problems, such as memory corruption, can occur that make it necessary for you to recover your configuration from a backed up version.

Note Each time that you enter a **copy running-config startup-config** command, a binary file is created and the ASCII file is updated. A valid binary configuration file reduces the overall boot time significantly. A binary file cannot be uploaded, but its contents can be used to overwrite the existing startup configuration. The **write erase** command clears the binary file.

SUMMARY STEPS

1. **write erase**
2. **reload**
3. **copy** *configuration_file* **running-configuration**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**write erase** **Example:** `switch# write erase`	Clears the current configuration of the switch.
Step 2	**reload** **Example:** `switch# reload`	Restarts the device. You will be prompted to provide a kickstart and system image file for the device to boot and run. **Note** By default, the **reload** command reloads the device from a binary version of the startup configuration. You can use the **reload ascii** command to copy an ASCII version of the configuration to the start up configuration when reloading the device.
Step 3	**copy** *configuration_file* **running-configuration** **Example:**	Copies a previously saved configuration file to the running configuration.

	Command or Action	Purpose
	`switch# copy bootflash:start-config.bak running-configuration`	**Note** The *configuration_file* filename argument is case sensitive.
Step 4	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the start-up configuration.

Removing the Configuration for a Missing Module

When you remove an I/O module from the chassis, you can also remove the configuration for that module from the running configuration.

Note You can only remove the configuration for an empty slot in the chassis.

Before you begin

Remove the I/O module from the chassis.

SUMMARY STEPS

1. (Optional) **show hardware**
2. **purge module** *slot* **running-config**
3. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **show hardware** **Example:** `switch# show hardware`	Displays the installed hardware for the device.
Step 2	**purge module** *slot* **running-config** **Example:** `switch# purge module 3 running-config`	Removes the configuration for a missing module from the running configuration.
Step 3	(Optional) **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Erasing a Configuration

You can erase the configuration on your device to return to the factory defaults.

You can erase the following configuration files saved in the persistent memory on the device:

- Startup
- Boot
- Debug

Note The **write erase** command erases the entire startup configuration, except for the following:

- Boot variable definitions
- The IPv4 configuration on the mgmt0 interface, including the following:
  - Address
  - Subnet mask

To remove the boot variable definitions and the IPv4 configuration on the mgmt0 interface, use the **write erase boot** command.

Important POAP is enabled by default, and it will prevent you from using the switch after you use the **write erase** and **reload** commands. To use the switch after erasing the configuration on the switch and reloading it, ensure that you have access to the console.

SUMMARY STEPS

1. **write erase** [**boot** | **debug**]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**write erase** [**boot**	**debug**]
		Example:
		`switch# write erase`
		`Warning: This command will erase the startup-configuration.`
		`Do you wish to proceed anyway? (y/n)  [n] y`
			Erases configurations in persistent memory. The default action erases the startup configuration.
			The **boot** option erases the boot variable definitions and the IPv4 configuration on the mgmt0 interface.
			The **debug** option erases the debugging configuration.
			Note The running configuration file is not affected by this command.

Clearing Inactive Configurations

You can clear inactive Quality of Service (QoS) and/or access control list (ACL) configurations.

SUMMARY STEPS

1. (Optional) **show running-config** *type* **inactive-if-config**

2. **clear inactive-config** *policy*
3. (Optional) **show inactive-if-config log**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **show running-config** *type* **inactive-if-config** **Example:** `# show running-config ipqos inactive-if-config`	Displays any inactive ACL or QoS configurations. The values for the *type* argument are **aclmgr** and **ipqos**. • **aclmgr**— Displays any inactive configurations for aclmgr. • **ipqos**—Displays any inactive configurations for qosmgr.

Step 2	**clear inactive-config** *policy*
	Example:
	`# clear inactive-config qos`
	`clear qos inactive config`
	`Inactive if config for QoS manager is saved at/bootflash/qos_inactive_if_config.cfg for vdc default & for other than default vdc: /bootflash/vdc_x/qos_inactive_if_config.cfg (where x is vdc number)`
	`you can see the log file @ show inactive-if-config log`
		Clears inactive configurations.
		The values for the *policy* argument are **qos** and **acl**.
		The following describes the values:
		• **qos**—Clears inactive QoS configurations.
		• **acl**— Clears inactive ACL configurations.
		• **acl qos**—Clears inactive ACL configurations and inactive QoS configurations.

Step 3	(Optional) **show inactive-if-config log** **Example:** `# show inactive-if-config log`	Displays the commands that were used to clear the inactive configurations.

Configuration Archive and Configuration Log

This section contains information on configuration archive and configuration log.

Information About Configuration Archive

The configuration archive is intended to provide a mechanism to store, organize, and manage an archive of the configuration files to enhance the configuration rollback capability provided by the **configure replace** command. Before configuration archiving was introduced, you could save copies of the running configuration using the **copy running-config** *destination-url* command, storing the replacement file either locally or remotely. However, this method lacked any automated file management. The configuration replace and configuration rollback provides the capability to automatically save copies of the running configuration to the configuration archive. These archived files serve as checkpoint configuration references and can be used by the **configure replace** command to revert to the previous configuration states.

The **archive config** command allows you to save configurations in the configuration archive using a standard location and filename prefix that is automatically appended with an incremental version number (and optional timestamp) as each consecutive file is saved. This functionality provides a means for consistent identification

of saved configuration files. You can specify how many versions of the running configuration are kept in the archive. After the maximum number of files are saved in the archive, the oldest file is automatically deleted when the next, most recent file is saved. The **show archive** command displays information for all configuration files saved in the configuration archive.

The configuration archive, wherein the configuration files are stored and are available for use with the **configure replace** command, can be located on the following file systems:

- If your platform has disk0--disk0:, disk1:, ftp:, pram:, rcp:, slavedisk0:, slavedisk1:, or tftp:
- If your platform does not have disk0--bootflash:, tftp:,, and ftp:

Note The TFTP and FTP for this feature use VRF management.

Configuring the Characteristics of the Configuration Archive

Before using the **archive config** command, the configuration archive must be configured. Complete the following steps to configure the characteristics of the configuration archive:

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **archive**
3. switch(config-archive)# **path** *url*
4. switch(config-archive)# **maximum** *number*
5. switch(config-archive)# **time-period** *minutes*
6. switch# **archive config**
7. switch#**show archive log config all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal** **Example:** `switch# configure terminal`	Enters the global configuration mode.
Step 2	switch(config)# **archive** **Example:** `switch(config)# archive`	Enters the archive configuration mode.
Step 3	switch(config-archive)# **path** *url* **Example:** `switch(config-archive)# path bootflash:myconfig`	Specifies the location and the filename prefix for the files in the configuration archive. • Depending on your hardware platform, the name of your file system can be different than the one displayed in the example.

	Command or Action	Purpose
		Note If a directory is specified in the path instead of the file, the directory name must be followed by a forward slash as follows: path flash:/*directory*/. The forward slash is not necessary after a filename; it is necessary only when specifying a directory.
Step 4	switch(config-archive)# **maximum** *number* **Example:** `switch(config-archive)# maximum 14`	(Optional) Sets the maximum number of archive files of the running configuration to be saved in the configuration archive. • The *number* is the maximum number of the archive files of the running configuration that can be saved in the configuration archive. The range is 1 to 14. The default is 10. **Note** Before using this command, you must configure the **path** to specify the location and filename prefix for the files in the configuration archive.
Step 5	switch(config-archive)# **time-period** *minutes* **Example:** `switch(config-archive)# time-period 10`	(Optional) Sets the time increment for automatically saving an archive file of the current running configuration in the configuration archive. • The *minutes* argument specifies how often, in minutes, to automatically save an archive file of the current running configuration in the configuration archive. **Note** Before using this command, you must configure the **path** command to specify the location and filename prefix for the files in the configuration archive.
Step 6	switch# **archive config** **Example:** `switch# archive config`	Saves the current running configuration file to the configuration archive. **Note** You must configure the **path** before using the **archive config** command.
Step 7	switch#**show archive log config all**	Displays the configuration log entries for all the users.

Information About Configuration Log

The configuration change notification and logging tracks the changes that are made to the running configuration by maintaining a configuration log. This configuration log tracks the changes that are initiated only through the CLI or HTTP. Only complete commands that result in the invocation of action routines are logged. The following types of entries are not logged:

- Commands that result in a syntax error message

- Partial commands that invoke the device help system

The configuration log tracks the changes that are initiated only through the CLI. For each configuration command that is executed, the following information is logged:

- A configuration change sequence number
- The line from which the command was executed
- The name of the user that executed the command
- The command that was executed

You can display the information from the configuration log by using the **show archive log config all** command

For each configuration command that is executed, the following information is logged:

- The command that was executed
- The configuration mode in which the command was executed
- The name of the user that executed the command
- The time at which the command was executed
- A configuration change sequence number
- Parser return codes for the command

You can display the information from the configuration log by using the **show archive log config** command, with the exception of the parser return codes, that are used for an internal purpose only.

Displaying Configuration Log Entries

To display the configuration log entries, the configuration change logging provides the **show archive log config all** command.

SUMMARY STEPS

1. switch# **show archive log config** *number* [*end-number*]
2. switch# **show archive log config user username**
3. switch# **show archive log config user username first-index start-number [last-index end-number]**

DETAILED STEPS

Step 1 switch# **show archive log config** *number* [*end-number*]

Displays the configuration log entries for all users

Example:

```
switch# show archive log config all

NDEX   LINE                    USER          LOGGED COMMAND
1       console0                user01       | logging console 1
2       console0                user01       | logging monitor 2
```

```
3       console0                user02       | system default switchport shutdown
4       console0                user02       | interface mgmt0
5       console0                user02       |  no shutdown
```

Step 2 switch# **show archive log config user username**

Displays the configuration log entries for the specified username.

Example:

The following example displays the configuration log entries for a specified username.

```
switch# show archive log config user user02

INDEX   LINE                    USER          LOGGED COMMAND
3       console0                user02       | system default switchport shutdown
4       console0                user02       | interface mgmt0
5       console0                user02       |  no shutdown
```

Step 3 switch# **show archive log config user username first-index start-number [last-index end-number]**

Displays the configuration log entries by the index numbers. If you specify a number for the optional last-index, all the log entries with the index numbers in the range from the value entered for the start-number through the end-number for the specified user are displayed.

Example:

The following example displays the configuration log entry numbers 4 and 5 for a user with the username, user02. The range for the first-index and last-index is 1 to 2000000000.

```
switch# show archive log config user user02  first-index 4 last-index 5
Last Log cleared/wrapped time is : Wed Oct 19 00:53:08 2016


INDEX   LINE                    USER          LOGGED COMMAND
4       console0                user02       | interface mgmt0
5       console0                user02       |  no shutdown
```

Verifying the Device Configuration

To verify the configuration after bootstrapping the device using POAP, use one of the following commands:

Command	Purpose
show running-config	Displays the running configuration.
show startup-config	Displays the startup configuration.
show time-stamp running-config last-changed	Displays the timestamp when the running configuration was last changed.

For detailed information about the fields in the output from these commands, see the Cisco Nexus command reference for your device.

Examples of Working with Configuration Files

This section includes examples of working with configuration files.

Copying Configuration Files

This example shows how to copy a running configuration to the bootflash: file system:

Backing Up Configuration Files

This example shows how to back up the startup configuration to the bootflash: file system (ASCII file):

```
switch# copy startup-config bootflash:my-config
```

This example shows how to back up the startup configuration to the TFTP server (ASCII file):

```
switch# copy startup-config tftp://172.16.10.100/my-config
```

This example shows how to back up the running configuration to the bootflash: file system (ASCII file):

```
switch# copy running-config bootflash:my-config
```

Rolling Back to a Previous Configuration

To roll back your configuration to a snapshot copy of a previously saved configuration, you need to perform the following steps:

1. Clear the current running image with the **write erase** command.
2. Restart the device with the **reload** command.

Note By default, the **reload** command reloads the device from a binary version of the startup configuration.

You can use the **reload ascii** command to copy an ASCII version of the configuration to the start up configuration when reloading the device.

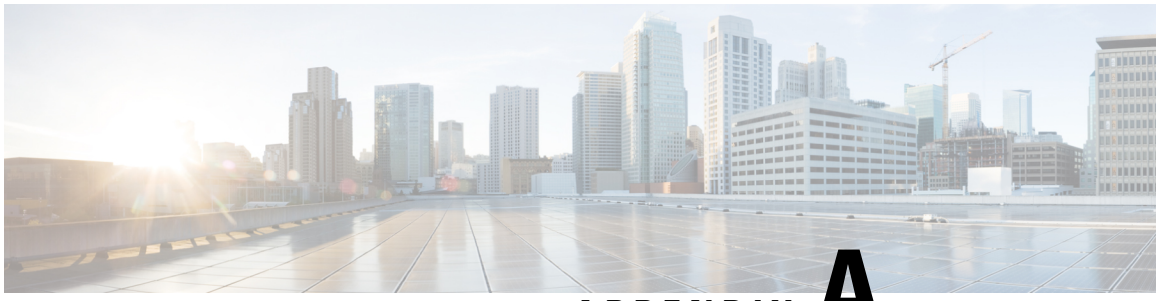
3. Copy the previously saved configuration file to the running configuration with the **copy** *configuration_file* **running-configuration** command.
4. Copy the running configuration to the start-up configuration with the **copy running-config startup-config** command.

Additional References for Configuration Files

This section includes additional information related to managing configuration files.

Related Documents for Configuration Files

Related Topic	Document Title
Licensing	*Cisco NX-OS Licensing Guide*
Command reference	*Cisco Nexus 3000 Series NX-OS Command Reference*



APPENDIX A

Supported Standards

Supported Standards

This table lists the IEEE compliance standards.

Table 12: IEEE Compliance Standards

Standard	Description
802.1D	MAC Bridges
802.1p	Class of Service Tagging for Ethernet frames
802.1Q	VLAN Tagging
802.1s	Multiple Spanning Tree Protocol
802.1w	Rapid Spanning Tree Protocol
802.3ab	1000Base-T (10/100/1000 Ethernet over copper)
802.3ad	Link aggregation with LACP
802.3ae	10-Gigabit Ethernet

This table lists the RFC compliance standards. For information on each RFC, see www.ietf.org.

Table 13: RFC Compliance Standards

Standard	Description
BGP	
RFC 1997	BGP Communities Attribute

Standard	Description
RFC 2385	Protection of BGP Sessions via the TCP MD5 Signature Option
RFC 2439	BGP Route flap damping
RFC 2519	A Framework for Inter-Domain Route Aggregation
RFC 2858	Multiprotocol Extensions for BGP-4
RFC 3065	Autonomous System Confederations for BGP
RFC 3392	Capabilities Advertisement with BGP-4
RFC 4271	BGP version 4
RFC 4273	BGP4 MIB - Definitions of Managed Objects for BGP-4
RFC 4456	BGP Route reflection
RFC 4486	Subcodes for BGP cease notification message
RFC 4724	Graceful Restart Mechanism for BGP
RFC 4893	BGP Support for Four-octet AS Number Space
ietf-draft	Bestpath transition avoidance (draft-ietf-idr-avoid-transition-05.txt)
ietf-draft	Peer table objects (draft-ietf-idr-bgp4-mib-15.txt)
ietf-draft	Dynamic Capability (draft-ietf-idr-dynamic-cap-03.txt)
IP Multicast	
RFC 2236	Internet Group Management Protocol, Version 2
RFC 3376	Internet Group Management Protocol, Version 3

Standard	Description
RFC 3446	Anycast Rendezvous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)
RFC 3569	An Overview of Source-Specific Multicast (SSM)
RFC 3618	Multicast Source Discovery Protocol (MSDP)
RFC 4601	ASM - Sparse Mode (PIM-SM): Protocol Specification (Revised)
RFC 4607	Source-Specific Multicast for IP
RFC 4610	Anycast-RP Using Protocol Independent Multicast (PIM)
RFC 6187	X.509v3 Certificates for Secure Shell Authentication
ietf-draft	Mtrace server functionality, to process mtrace-requests, draft-ietf-idmr-traceroute-ipm-07.txt
IP Services	
RFC 768	UDP
RFC 783	TFTP
RFC 791	IP
RFC 792	ICMP
RFC 793	TCP
RFC 826	ARP
RFC 854	Telnet
RFC 959	FTP
RFC 1027	Proxy ARP
RFC 1305	NTP v3
RFC 1519	CIDR
RFC 1542	BootP relay
RFC 1591	DNS client

Standard	Description
RFC 1812	IPv4 routers
RFC 2131	DHCP Helper
RFC 2338	VRRP
IS-IS	
RFC 1142 (OSI 10589)	OSI 10589 Intermediate system to intermediate system intra-domain routing exchange protocol
RFC 1195	Use of OSI IS-IS for routing in TCP/IP and dual environment.
RFC 2763	Dynamic Hostname Exchange Mechanism for IS-IS
RFC 2966	Domain-wide Prefix Distribution with Two-Level IS-IS
RFC 2973	IS-IS Mesh Groups
RFC 3277	IS-IS Transient Blackhole Avoidance
RFC 3373	Three-Way Handshake for IS-IS Point-to-Point Adjacencies
RFC 3567	IS-IS Cryptographic Authentication
RFC 3847	Restart Signaling for IS-IS
ietf-draft	Internet Draft Point-to-point operation over LAN in link-state routing protocols (draft-ietf-isis-igp-p2p-over-lan-06.txt)
OSPF	
RFC 2328	OSPF Version 2
RFC 2370	OSPF Opaque LSA Option
RFC 2740	OSPF for IPv6 (OSPF version 3)
RFC 3101	OSPF Not-So-Stubby-Area (NSSA) Option
RFC 3137	OSPF Stub Router Advertisement
RFC 3509	Alternative Implementations of OSPF Area Border Routers

Standard	Description
RFC 3623	Graceful OSPF Restart
RFC 4750	OSPF Version 2 MIB
RIP	
RFC 1724	RIPv2 MIB extension
RFC 2082	RIPv2 MD5 Authentication
RFC 2453	RIP Version 2

Supported Standards