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

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.
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.

	Command or Action	**Purpose**

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:**	Displays the running configuration.

	Command or Action	**Purpose**
	`switch# show running-config`	
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*
		
		Example:
		`switch# copy startup-config`
		`bootflash:sw1-start-config.bak`
			Copies the startup-configuration file to internal memory.
			
			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:** `switch# copy bootflash:start-config.bak running-configuration`	Copies a previously saved configuration file to the 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.

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.

	Command or Action	Purpose
		• **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 icon]

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. **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.

	Command or Action	**Purpose**
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.

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	