Working with Configuration Files

This chapter contains the following sections:

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 2

Downloading the Startup Configuration From a Remote Server, on page 5

Licensing Requirements for Configuration Files

The following table shows the licensing requirements for this feature:

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

Managing Configuration Files

This section describes how to manage configuration files.

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	**show running-config** **Example:** `switch# show running-config`	(Optional) Displays the running configuration.

	Command or Action	Purpose
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:**, or **sftp:**. 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:**, or **sftp:**. 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.

This example shows how to copy the bootflash file using FTP:

```
switch# copy ftp: bootflash:
Enter source filename: n5000-uk9-kickstart.5.0.2.N2.1.bin
Warning: There is already a file existing with this name. Do you want to
overwrite (y/n)?[n] y
Enter vrf (If no input, current vrf 'default' is considered): management
Enter hostname for the ftp server: 172.1.1.10
Enter username: xxx
Password:
***** Transfer of file Completed Successfully *****
Note: Boot variable kickstart is set to
bootflash:/n5000-uk9-kickstart.5.0.2.N2.1.bin
```

This example shows how to copy the bootflash file using FTP:

```
switch# copy ftp: bootflash:
Enter source filename: n5500-uk9-kickstart.5.0.2.N2.1.bin
Warning: There is already a file existing with this name. Do you want to
overwrite (y/n)?[n] y
Enter vrf (If no input, current vrf 'default' is considered): management
Enter hostname for the ftp server: 172.1.1.10
Enter username: xxx
Password:
***** Transfer of file Completed Successfully *****
Note: Boot variable kickstart is set to
bootflash:/n5500-uk9-kickstart.5.0.2.N2.1.bin
```

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:**, or **sftp:**. 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	**show running-config** **Example:** `switch# show running-config`	(Optional) Displays the running configuration.

	Command or Action	Purpose
Step 3	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.
Step 4	**show startup-config** **Example:** `switch# show startup-config`	(Optional) 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), or Secure Shell FTP (SFTP) 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>`
	`...`
	`Would you like to enter the basic configuration`
	
	`dialog (yes/no): n`
	`switch#`
		Reloads the Cisco NX-OS device.
		Note Do not use the setup utility to configure the device.

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:**, or **sftp:**. 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	**show startup-config** **Example:** `switch# show startup-config`	(Optional) 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	**dir** {**slot0:** | **usb1:** | **usb2:**}[*directory/*] **Example:** `switch# dir slot0:`	(Optional) 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	**dir** {**slot0:** | **usb1:** | **usb2:**}[*directory/*] **Example:** `switch# dir slot0:`	(Optional) 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	**show running-config** **Example:** `switch# show running-config`	(Optional) Displays the running configuration.
Step 4	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.
Step 5	**show startup-config** **Example:** `switch# show startup-config`	(Optional) Displays the startup configuration.

Related Topics

Copying Files

Copying the Startup Configuration from an External Flash Memory Device

You can recover the startup configuration on your Cisco NX-OS device by downloading a new startup configuration file saved on 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* **startup-config**
3. (Optional) **show startup-config**

DETAILED STEPS

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

	Command or Action	Purpose
	Example: `switch# copy running-config bootflash:sw1-run-config.bak`	
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.

	Command or Action	Purpose
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. Beginning with Cisco NX-OS 6.2(2), 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	**show hardware** **Example:** `switch# show hardware`	(Optional) 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	**copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	(Optional) 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	**show running-config** *type* **inactive-if-config** **Example:** `# show running-config ipqos inactive-if-config`	(Optional) 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)`
		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.

	Command or Action	Purpose
	`you can see the log file @ show inactive-if-config log`	
Step 3	**show inactive-if-config log** **Example:** `# show inactive-if-config log`	(Optional) Displays the commands that were used to clear the inactive configurations.

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.

Beginning with Cisco NX-OS 6.2(2), 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	