



CHAPTER 26

Working With the Flash File System

This chapter describes how to use the flash file system on the Catalyst 6500 series switches.



Note

For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

- [Understanding How the Flash File System Works, page 26-1](#)
- [Working with the Flash File System on the Switch, page 26-2](#)

Understanding How the Flash File System Works

The flash file system on a Catalyst 6500 series supervisor engine provides a number of useful commands to help you manage the software image and configuration files. The flash file system on the supervisor engine consists of the flash devices on which you can store the files:

- Supervisor Engine 1 and Supervisor Engine 2
 - **bootflash:** Onboard flash memory
 - **slot0:** Linear Flash PC card (PCMCIA slot)
 - **disk0:** ATA Flash PC card (PCMCIA slot)
- Supervisor Engine 720
 - **bootflash:** Onboard flash memory
 - **disk0:** CompactFlash Type II card only (disk 0 slot)
 - **disk1:** CompactFlash Type II card (disk 1 slot)
- Supervisor Engine 32
 - **bootdisk:** Onboard flash memory
 - **disk0:** CompactFlash Type II card only (disk 0 slot)

Working with the Flash File System on the Switch

These sections describe how to work with the flash file system:

- [Setting the Default Flash Device, page 26-2](#)
- [Setting the Text File Configuration Mode, page 26-2](#)
- [Setting the Text File Configuration Mode to Auto-Save, page 26-3](#)
- [Listing the Files on a Flash Device, page 26-5](#)
- [Copying Files, page 26-6](#)
- [Deleting Files, page 26-8](#)
- [Restoring Deleted Files, page 26-8](#)
- [Verifying a File Checksum, page 26-9](#)
- [Formatting a Flash Device, page 26-9](#)

Setting the Default Flash Device

When you set the default flash device for the switch, the default device is assumed when you enter a flash file system command without specifying the flash device.

To set the default flash device, perform this task:

	Task	Command
Step 1	Set the default flash device for the switch.	<code>cd [[m/][bootflash: slot0:]]</code>
Step 2	Verify the default flash device for the switch.	<code>pwd [mod]</code>

This example shows how to change the default flash device to **slot0:** and verify the default device:

```
Console> (enable) cd slot0:
Console> (enable) pwd
slot0
Console> (enable)
```

Setting the Text File Configuration Mode

When you use text file configuration mode, the switch stores its configuration as a text file in nonvolatile storage, either in NVRAM or flash memory. This text file consists of the commands that are entered by you to configure the various features. For example, if you disable a port, the command to disable that port will be in the text configuration file.

Because the text file only contains the commands that you have used to configure your switch, it typically uses less NVRAM or flash memory space than binary configuration mode. Because the text file in most cases requires less space, NVRAM is a good place to store the file. If the text file exceeds NVRAM space, it can also be saved to flash memory.

When operating in text file configuration mode, most user settings are not immediately saved to NVRAM; the configuration changes are only written to DRAM. You will need to enter the **write memory** command to store the configuration in nonvolatile storage.

**Note**

The VLAN commands are not saved as part of the configuration file when the switch is operating in text mode with the VTP mode set to server.

To set the text file configuration mode, perform this task:

	Task	Command
Step 1	Set the file configuration mode for the system to text.	set config mode text { <i>nvramp device:file-id</i> }
Step 2	Verify the file configuration mode for the system.	show config mode
Step 3	Save the text file configuration.	write memory
Step 4	Display the current run-time configuration.	show running-config all
Step 5	Display the startup configuration that will be used after the next reset.	show config

This example shows how to configure the system to save its configuration as a text file in NVRAM, verify the configuration mode, and display the current run-time configuration:

```
Console> (enable) set config mode text nvramp
Console> (enable) show config mode
Console> (enable) show running-config all
Console> (enable) show config
Console> (enable)
```

Setting the Text File Configuration Mode to Auto-Save

Use the **set config mode text auto-save** command to save the text configuration in NVRAM automatically. Use the **interval** keyword to set the time interval between the occurrences of saving the text configuration in NVRAM. You can specify the time interval between the occurrences of saving the text configuration in NVRAM even if the system is in binary mode. If you do not specify the number of minutes after entering the **interval** keyword, the interval is set to the default of 30 minutes.

**Note**

In software release 8.4(1) and earlier releases, valid values for the *mins* argument are from 30 minutes to 35000 minutes. In release 8.4(2) and subsequent releases, valid values for the *mins* argument are from 1 minute to 35000 minutes.

The text configuration is not saved automatically in NVRAM unless auto-save is enabled. To enable auto-save, you must first set the system configuration mode to text and configure the system to save the text configuration in NVRAM. If the system configuration mode is set to binary mode, you cannot enable auto-save.

To set the text file configuration mode to auto-save, perform this task:

	Task	Command
Step 1	Set the file configuration mode for the system to text.	set config mode text {nvram device:file-id}
Step 2	Specify the auto-save keyword.	set config mode text auto-save {enable disable}
Step 3	(Optional) Configure the interval keyword.	set config mode text auto-save interval mins
Step 4	Verify the file configuration mode for the system.	show config mode
Step 5	Save the text file configuration.	write memory
Step 6	Display the current run-time configuration.	show running-config all
Step 7	Display the startup configuration that will be used after the next reset.	show config

This example shows how to set the configuration mode to text and designate the location and filename for saving the text configuration file:

```
Console> (enable) set config mode text bootflash:switch.cfg
Binary system configuration has been deleted from NVRAM. Configuration mode set to text.
Use the write memory command to save configuration
changes. System configuration file set to: bootflash:switch.cfg
The file specified will be used for configuration during the next bootup.
Console> (enable)
```

This example shows how to enable auto-save when the configuration is set to text mode and the system is configured to save the text configuration in NVRAM:

```
Console> (enable) set config mode text auto-save enable
auto-save feature has been enabled
auto-save feature has started
Please do a write mem manually if you plan to reboot the switch or any card before first
expiry of the timer
Console> (enable)
```

This example shows the message that is displayed if you attempt to enable auto-save when the configuration is not set to text mode and the system is not configured to save the text configuration in NVRAM:

```
Console> (enable) set config mode text auto-save enable
auto-save cannot be enabled unless config mode is set to text and config file is stored in
nvram.
Use the 'set config mode text nvram' command to enable automatic saving of the system
configuration to nvram
Console> (enable)
```

This example shows how to set the interval between the saves to 2880 minutes:

```
Console> (enable) set config mode text auto-save interval 2880
auto-save interval set to 2880 minutes
Console> (enable)
```

This example shows how to set the interval between the saves to the default setting of 30 minutes:

```
Console> (enable) set config mode text auto-save interval
auto-save interval set to 30 minutes
Console> (enable)
```

Listing the Files on a Flash Device

To list the files on a flash device, perform one of these tasks:

Task	Command
Display a list of files on a flash device.	dir <i>[[m/]device:][filename]</i>
Display a list of the deleted files on a flash device.	dir <i>[[m/]device:][filename]</i> deleted
Display a list of all files on a flash device, including the deleted files.	dir <i>[[m/]device:][filename]</i> all
Display a detailed list of files on a flash device.	dir <i>[[m/]device:][filename]</i> long

This example shows how to list the files on the default flash device:

```
Console> (enable) dir
-#- -length- -----date/time----- name
   4 3134688 Mar 15 1999 08:27:01 cat6000-sup.5-2-1-CSX.bin
   5 3231989 Jan 24 1999 12:04:40 cat6000-sup.5-1-1-CSX.bin
   6      135 Feb 17 1999 11:30:05 dns_config.cfg

1213952 bytes available (6388224 bytes used)
Console> (enable)
```

This example shows how to list the files on a flash device other than the default device:

```
Console> (enable) dir slot0:
-#- -length- -----date/time----- name
   1 3209261 Jun 16 1998 13:18:19 cat6000-sup.5-2-1-CSX.bin
   2      135 Jul 17 1998 11:32:53 dns-config.cfg
   3 3231989 Jul 17 1998 16:54:23 cat5000-sup3.4-1-2.bin
   4      8589 Jul 17 1998 17:02:52 6000_config.cfg

9933504 bytes available (6450496 bytes used)
Console> (enable)
```

This example shows how to list the deleted files on the default flash device:

```
Console> (enable) dir deleted
-#- ED --type-- --crc--- -seek-- nlen -length- -----date/time----- name
   1 .D ffffffff 81a027ca 41bdc 22 7004 Apr 01 1998 15:27:45 5002.config.
4.1.98.cfg
   2 .D ffffffff ccce97a3 43644 23 6630 Apr 01 1998 15:36:47 5002.default
.config.cfg
   3 .D ffffffff 81a027ca 45220 15 7004 Apr 19 1998 10:05:59 5002_config.
cfg

1213952 bytes available (6388224 bytes used)
Console> (enable)
```


This example shows how to download a configuration file from a TFTP server for storage on a flash device:

```

Console> (enable) copy tftp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? dns-config.cfg
Flash device [slot0]?
Name of file to copy to [dns-config.cfg]?

9932056 bytes available on device slot0, proceed (y/n) [n]? y
/
File has been copied successfully.
Console> (enable)

```

This example shows how to copy the running configuration to flash memory:

```

Console> (enable) copy config flash
Flash device [bootflash]? slot0:
Name of file to copy to []? 6000_config.cfg

Upload configuration to slot0:6000_config.cfg
9942096 bytes available on device slot0, proceed (y/n) [n]? y
.....
.....
.....

.....
.....
..

Configuration has been copied successfully.
Console> (enable)

```

This example shows how to upload a configuration file on a flash device to a TFTP server:

```

Console> (enable) copy slot0:6000_config.cfg tftp
IP address or name of remote host []? 172.20.52.3
Name of file to copy to [6000_config.cfg]?
/
File has been copied successfully.
Console> (enable)

```

This example shows how to upload an image from a remote host into flash using rcp:

```

Console> (enable) copy rcp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? 6000_config.cfg
Flash device [bootflash]?
Name of file to copy to [6000_config.cfg]?

4369664 bytes available on device bootflash, proceed (y/n) [n]? y
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCC
File has been copied successfully.
Console> (enable)

```

Deleting Files



Caution

If you enter the **squeeze** command on a flash device, you cannot restore the files that were deleted prior to the **squeeze** command.

To delete the files on a flash device, perform this task in privileged mode:

	Task	Command
Step 1	Delete a file on a flash device.	delete <i>[[m/]device:]filename</i>
Step 2	If desired, permanently remove all deleted files on the flash device (this operation can take several minutes to complete).	squeeze <i>[m/]device:</i>
Step 3	Verify that the files are deleted.	dir <i>[[m/]device:]filename</i>

This example shows how to delete a file from a flash device:

```
Console> (enable) delete dns_config.cfg
Console> (enable)
```

This example shows how to remove all deleted files from a flash device permanently:

```
Console> (enable) squeeze slot0:
All deleted files will be removed, proceed (y/n) [n]? y
Squeeze operation may take a while, proceed (y/n) [n]? y
Erasing squeeze log
Console> (enable)
```

Restoring Deleted Files

You must specify the index number of a deleted file to identify the file to undelete. The index number for each file appears in the first column of the **dir** command output. A file cannot be undeleted if a valid file with the same name already exists. Instead, you must delete the existing file and then undelete the desired file. A file can be deleted and undeleted up to 15 times.

To restore the deleted files on a flash device, perform this task in privileged mode:

	Task	Command
Step 1	Identify the index number of the deleted files on the flash device.	dir <i>[[m/]device:]filename</i> deleted
Step 2	Restore a file on a flash device.	undelete <i>index</i> <i>[[m/]device:]</i>
Step 3	Verify that the file is restored.	dir <i>[[m/]device:]filename</i>

This example shows how to restore a deleted file:

```
Console> (enable) dir deleted
-#- ED --type-- --crc--- -seek-- nlen -length- -----date/time----- name
6 .D ffffffff 42da7f71 657a00 14 135 Jul 17 1998 11:30:05 dns_config.cfg

1213952 bytes available (6388224 bytes used)
Console> (enable) undelete 6
```

```

Console> (enable) dir
-#- -length- -----date/time----- name
   4  3134688 Apr 27 1998 08:27:01 cat6000-sup.5-2-1.bin
   5  3231989 Jun 24 1998 12:04:40 cat6000-sup.5-2-1.bin
   6      135 Jul 17 1998 11:30:05 dns_config.cfg

1213952 bytes available (6388224 bytes used)
Console> (enable)

```

Verifying a File Checksum

To verify the checksum of a file on a flash device, perform this task in privileged mode:

Task	Command
Verify the checksum of a file on a flash device.	verify <i>[[m/]device:] filename</i>

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

```

Console> (enable) verify cat6000-sup.5-2-1-CSX.bin
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCC
File bootflash:cat6000-sup.5-2-1-CSX.bin verified OK
Console> (enable)

```

Formatting a Flash Device

Some flash devices require formatting before they can be used. You can reserve up to 16 spare sectors for use when other sectors fail (by default, none are reserved). If you do not reserve spare sectors and later some sectors fail, you will have to reformat the entire flash memory, which erases all existing data.



Note

Supervisor Engine 2 and Supervisor Engine 1 do not support the same Flash PC card format. To use a Flash PC card with Supervisor Engine 2, format the card with Supervisor Engine 2. To use a Flash PC card with Supervisor Engine 1, format the card with Supervisor Engine 1.



Note

The Flash PC cards that are formatted on Supervisor Engine 1 or on a route-switch processor (RSP)-based Cisco 7500 series router are interchangeable if the router is running software at least at the same level as the supervisor engine. You cannot use the Flash PC cards that are formatted on a route processor (RP)-based Cisco 7000 series router without reformatting.

When you format a flash device, you can specify the *monlib* file (the ROM monitor library), which the ROM monitor uses to access the files in the flash file system. The *monlib* file is also compiled into the software image.

In the **format** command syntax, use the *device2* argument to specify the device that contains the monlib file to use. If you omit the entire *device2* argument, the switch formats the device using the monlib file that is bundled with the software. If you omit just the device name (*device2*) from the `[[device2:][monlib-filename]]` argument, the switch formats the device using the named monlib file from the default flash device. If you omit the *monlib-filename* from the `[[device2:][monlib-filename]]` argument, the switch formats the device using the monlib file from *device2*. If you specify the entire `[[device2:][monlib-filename]]` argument, the switch formats the device using the specified monlib file from the specified device. If the switch cannot find a monlib file, it terminates the formatting process.

**Note**

If the flash device has a volume ID, you must provide the volume ID to format the device. The volume ID is displayed using the **show flash m/device: filesystem** command.

To format a flash device, perform this task in privileged mode:

Task	Command
Format a flash device.	format [<i>spare spare-number</i>] [<i>mf</i>] <i>device1</i> : [[<i>device2</i> :] [<i>monlib-filename</i>]]

This example shows how to format the flash device in slot0:

```
Console> (enable) format slot0:
All sectors will be erased, proceed (y/n) [n]?y
Enter volume id (up to 31 characters):
Formatting sector 1
Format device slot0 completed.
Console> (enable)
```