



## Maintaining Router Memory

---

This chapter describes how to maintain and use the different types of memory on your router.

For a complete description of the memory commands mentioned in this chapter, refer to the “Router Memory Commands” chapter in the *Cisco IOS Configuration Fundamentals Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.



**Note**

Some of the commands mentioned in this chapter replace commands used in earlier Cisco IOS software releases. These replacements are listed in Table 14 for the benefit of those who may be accustomed to the older forms of these commands.

**Table 14 Mapping Old Commands to New Commands**

Old Command	New Command
copy erase flash	<b>erase flash:</b> (Class B Flash file systems only) <b>format</b> (Class A and C Flash file systems only)
copy verify or copy verify flash verify flash	<b>verify flash:</b> verify
copy verify bootflash verify bootflash	<b>verify bootflash:</b>

## Maintain Router Memory Task List

You can perform the tasks related to Flash memory in the following sections:

- Displaying Memory Information
- Comparing Types of Memory
- Reallocating DRAM Memory (Cisco 3600 series only)
- Partitioning Flash Memory
- Using Flash Load Helper to Upgrade Software on Run-from-Flash Systems
- Formatting Flash Memory

Format Flash Memory is a required first task if you are using a new PCMCIA Flash memory card on the Cisco 7000 family.

**Note**

These tasks assume you have a minimal configuration that you want to modify.

## Displaying Memory Information

Use the following commands in EXEC mode to display information about system memory:

Command	Purpose
<code>show flash-filesystem: [all   chips   filesystems]</code> (Class A Flash file systems) <code>show flash-filesystem: [partition number] [all   chips   detailed   err   summary]</code> (Class B Flash file systems) <code>show flash-filesystem:</code> (Class C Flash file systems)	Lists information about Flash memory.
<code>show file systems</code>	Lists the names of the file systems currently supported on the router.

Refer to the Release 12.1 *Cisco IOS Configuration Fundamentals Command Reference* for examples of these commands.

## Comparing Types of Memory

Your router has many different locations where it can store images, configuration files, and microcode. Refer to your hardware documentation for details on the following:

- Which types of memory your router contains
- Where files can be located
- Where images and boot images are located by default

### DRAM

Dynamic random-access memory contains two types of memory:

- Primary, main, or processor memory, which is reserved for the CPU to execute Cisco IOS software and to hold the running configuration and routing tables.
- Shared, packet, or I/O memory, which buffers data transmitted or received by the router's network interfaces.

On the Cisco 3600 series routers, you can use the **memory-size iomem** command to configure the proportion of DRAM devoted to main memory and to shared memory.

## EPROM

Erasable programmable read-only memory (EPROM). This memory is often referred to simply as ROM. It sometimes contains the following:

- ROM Monitor, which provides a user interface when the router cannot find a valid image.
- The boot loader/helper software (also called the boot image), which helps the router boot when it cannot find a valid Cisco IOS image in Flash memory.

## NVRAM

NVRAM stores the following information:

- Startup configuration file for every platform except Class A Flash file system platforms.
- For Class A Flash file system platforms, the location of the startup configuration depends on the CONFIG\_FILE Environment Variable.
- The software configuration register, which is used to determine which image to use when booting the router.

## Flash Memory

Flash memory stores the Cisco IOS software image. On some platforms, it can store configuration files or boot images.

Depending on the hardware platform, Flash memory might be available as EPROMs, single in-line memory modules (SIMMs), or Flash memory cards. Check the appropriate hardware installation and maintenance guide for information about types of Flash memory available on a specific platform.

Depending on the platform, Flash memory is available in the following forms:

- Internal Flash memory
  - Internal Flash memory often contains the system image.
  - Some platforms have two or more banks of Flash memory on one single in-line memory modules (SIMM). If the SIMM has two banks, it is sometimes referred to as *dual-bank Flash memory*. The banks can be partitioned into separate logical devices. See the “Partitioning Flash Memory” section for information about how to partition Flash memory.
- Bootflash
  - Bootflash often contains the boot image.
  - Bootflash sometimes contains the ROM Monitor.
- Flash memory PC cards or PCMCIA cards

A Flash memory card that is inserted in to a Personal Computer Memory Card International Association (PCMCIA) slot. This card is used to store system images, boot images, and configuration files.

The following platforms contains PCMCIA slots:

- The Cisco 1600 series routers include one PCMCIA slot.
- The Cisco 3600 series routers include two PCMCIA slots.

- The Cisco 7200 series Network Processing Engine (NPE) contains two PCMCIA slots
- The Cisco 7000 RSP7000 card and the Cisco 7500 series Route Switch Processor (RSP) card contain two PCMCIA slots.

Because the Cisco 3600 series and Cisco 7000 family can boot images and load configuration files from several locations, these systems use special ROM monitor environment variables to specify the location and filename of images and configuration files that the router is to use for various functions.

Some ciscoFlash MIB variables support the Flash file system on the Cisco 7000 family.

Note that the internal Flash and the Flash memory card cannot be used as a contiguous bank of Flash memory.

## Write Protection

Flash memory provides write protection against accidental erasing or reprogramming.

- Some platforms have a write-protect jumper which can be removed to prevent reprogramming of Flash memory. You must install the jumper when programming is required.
- Some platforms have write protect switched on Flash memory cards that you can use to protect data. You must set the switch to *unprotected* to write data to the Flash memory card.

Refer to your hardware documentation for information on security jumpers and write protect switches.

## Running from Flash Systems

Many Cisco routers load the system image from flash storage into RAM in order to run the Cisco IOS. However, some platforms, such as the Cisco 1600 Series and Cisco 2500 Series, execute the Cisco IOS directly in Flash memory. These platforms are run-from-Flash memory systems.

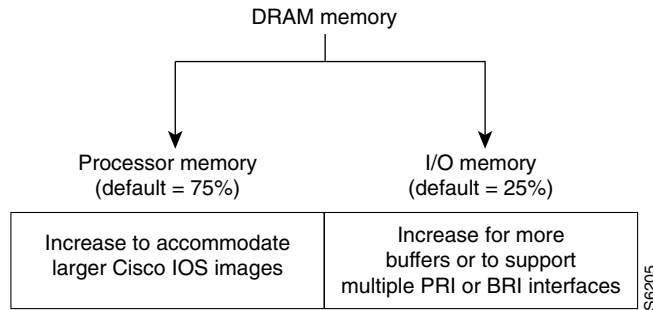
If you want to partition Flash memory, you must use a relocatable image. Relocatable images can be run from any location in Flash and can download images to any location. If you are upgrading from a nonrelocatable image to a relocatable image, you must erase Flash memory during the download so that the image is downloaded as the first file in Flash memory. All images for run-from-Flash platforms from Cisco IOS Release 11.0 and later are relocatable. See the “Image Naming Conventions” section in the “Loading and Maintaining System Images and Microcode” chapter to determine if your images are run-from-Flash images or are relocatable.

## Reallocating DRAM Memory (Cisco 3600 series only)

DRAM memory in Cisco 3600 series routers is organized as one contiguous address space divided between processor memory and I/O memory. Depending on the type and number of network interfaces you have configured in the router, you may need to reallocate the DRAM memory partitioned to processor memory and I/O memory.

Cisco manufacturing configures most Cisco 3600 series routers to have 25 percent of the address space allocated to I/O memory and 75 percent allocated to processor memory. But for customer orders that require two or more ISDN PRI interfaces, DRAM memory is configured to provide 40 percent of the address space for I/O memory and 60 percent for processor memory. (See Figure 10.) Cisco Systems performs these DRAM memory adjustments before it ships each router.

**Figure 10 Components and Uses of DRAM Memory for Cisco 3600 Series Routers**



**Note**

Routers running two or more ISDN PRI interfaces or 12 or more ISDN BRI interfaces require a DRAM memory configuration of 40 percent I/O memory and 60 percent processor memory.

However, there are cases where you may have to manually reallocate the DRAM memory split between processor memory and I/O memory after you have received a router from Cisco Systems.

For example, suppose you receive a Cisco 3640 router with the following running configuration:

- 2 Ethernet and 2 WAN interface card
- 8-port ISDN BRI with an NT1 network module
- IP feature set
- 16 MB of DRAM memory (by default, processor memory = 75%, I/O memory = 25%)
- 4 MB of Flash memory

Later, however, you add a 4-port ISDN BRI network module to the router. You now have 12 ISDN BRI interfaces running on the router. At this point, you must use the **memory-size iomem** command to configure 40 percent of the address space for I/O memory and 60 percent for processor memory.

To view your current mix of processor and I/O memory and reassign memory distribution accordingly, use the following commands beginning in privileged EXEC mode:

	<b>Command</b>	<b>Purpose</b>
<b>Step 1</b>	<code>show version</code>	View the total amount of memory loaded on the router.
<b>Step 2</b>	<code>show memory<sup>1</sup></code>	Determine the amount of free memory.
<b>Step 3</b>	<code>configure terminal</code>	Enter global configuration mode.
<b>Step 4</b>	<code>memory-size iomem I/O-memory-percentage<sup>2</sup></code>	Allocate processor memory and I/O memory.
<b>Step 5</b>	<code>exit</code>	Exit global configuration mode.
<b>Step 6</b>	<code>copy system:running-config nvram:startup-config</code>	Save the configuration to NVRAM.
<b>Step 7</b>	<code>reload</code>	Reload the router to run the new image.

1. The Free(b) column in the **show memory** command's output shows how much I/O memory is available.  
 2. The default is 40 percent for I/O memory and 60 percent for processor memory.

Valid I/O memory percentage values are 10, 15, 20, 25, 30, 40 (the default), and 50. I/O memory size is the specified percentage of total memory size, rounded down to the nearest multiple of 1 MB. A minimum of 4 MB of memory is required for I/O memory. The remaining memory is processor memory.

The **memory-size iomem** command does not take effect until you save it to NVRAM using the **copy system:running-config nvram:startup-config EXEC** command and reload the router. However, when you enter the command, the software checks whether the new memory distribution leaves enough processor memory for the currently running Cisco IOS image. If not, the following message appears:

```
Warning: Attempting a memory partition that does not provide enough Processor memory for
the current image.If you write memory now, this version of software may not be able to
run.
```

When you enter the **reload** command to run a new image, the software calculates the new processor and I/O memory split. If there is not enough processor memory, it automatically reduces I/O memory to an alternative setting to load the image. If there is still not enough processor memory for the image to run, then you do not have enough DRAM.

## Reallocate Processor Memory and I/O Memory Examples

The following example allocates 40 percent of DRAM to I/O memory and the remaining 60 percent to processor memory. The example views the current allocation of memory, changes the allocation, saves the allocation, and reloads the router so the changes can take effect. In the **show memory** command output, the Free(b) column shows how much I/O memory is available:

```
Router# show memory
          Head   Total (b)   Used(b)   Free (b)   Lowest (b)   Largest (b)
Processor 60913730  3066064    970420    2095644    2090736     2090892
         I/O   C00000    4194304   1382712    2811592     2811592     2805492
--More--

Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# memory-size iomem 40
Router(config)# exit
Router#
Router# copy system:running-config nvram:startup-config
Building configuration...
[OK]

Router# reload

rommon > boot
program load complete, entry point: 0x80008000, size: 0x32ea24
Self decompressing the image :
#####
#####
##### [OK]
```

## Partitioning Flash Memory

On most Class B Flash file systems, you can partition banks of Flash memory into separate, logical devices so that the router can hold and maintain two or more different software images. This partitioning allows you to write software into Flash memory while running software in another bank of Flash memory.

## Systems that Support Partitioning

To partition Flash memory, you must have at least two banks of Flash memory; a bank is a set of 4 chips. This requirement includes systems that support a single SIMM that has two banks of Flash memory. The minimum partition size is the size of a bank.

CiscoFlash MIB variables support partitioned Flash.

## Benefits of Partitioning Flash Memory

Partitioning Flash memory provides the following benefits:

- For any system, partitioning—rather than having one logical Flash memory device—provides a cleaner way of managing different files in Flash memory, especially if the Flash memory size is large.
- For systems that execute code out of Flash memory, partitioning allows you to download a new image into the file system in one Flash memory bank while an image is being executed from the file system in the other bank. The download is simple and causes no network disruption or downtime. After the download is complete, you can switch over to the new image at a convenient time.
- One system can hold two different images, one image acting as a backup for the other. Therefore, if a downloaded image fails to boot for some reason, the earlier running, good image is still available. Each bank is treated as a separate device.

## Flash Load Helper Versus Dual Flash Bank

Flash load helper is a software option that enables you to upgrade system software on run-from-Flash systems that have a single bank of Flash memory. It is a lower-cost software upgrade solution than dual-bank Flash, which requires two banks of Flash memory on one SIMM. Flash load helper is only available on run-from-Flash platforms, such as the Cisco 2500 series, Cisco 3000, and Cisco 5200.

You might use Flash load helper rather than partitioning Flash into two banks for one of the following reasons:

- If you want to download a new file into the same bank from which the current system image is executing.
- If you want to download a file that is larger than the size of a bank, and hence want to switch to a single-bank mode.
- If you have only one single-bank Flash SIMM installed. In this case, Flash load helper is the best option for upgrading your software.

See the “Using Flash Load Helper to Upgrade Software on Run-from-Flash Systems” section for information about using Flash load helper.

## Partition Flash Memory

To partition Flash memory, use the following command in global configuration mode:

Command	Purpose
<pre>partition flash partitions [size1 size2] partition flash-filesystem: [number-of-partitions][partition-size] (Cisco 1600 series and Cisco 3600 series)</pre>	Partition Flash memory.

This task will succeed only if the system has at least two banks of Flash and the partitioning does not cause an existing file in Flash memory to be split across the partitions.

For all platforms except the Cisco 1600 series and Cisco 3600 series, Flash memory can only be partitioned into two partitions.

For the Cisco 1600 series and Cisco 3600 series, the number of partitions that you can create in a Flash memory device equals the number of banks in the device. Enter the **show flash-filesystem: all** command to view the number of banks on the Flash memory device. The number of partition size entries you set must be equal to the number of specified partitions. For example, the **partition slot0: 2 8 8** command configures two partitions to be 8 MB in size each. The first 8 corresponds to the first partition; the second 8 corresponds to the second partition.

## Using Flash Load Helper to Upgrade Software on Run-from-Flash Systems

Flash load helper is a software option that enables you to upgrade system software on run-from-Flash systems that have a single bank of Flash memory. It is a lower-cost software upgrade solution than dual-bank Flash, which requires two banks of Flash memory on one SIMM.

The Flash load helper software upgrade process is simple and does not require additional hardware; however, it does require some brief network downtime. A system image running from Flash can use Flash load helper only if the boot ROMs support Flash load helper. Otherwise, you must perform the Flash upgrade manually. See the “Manually Boot from Flash Memory” section.

Flash load helper is an automated procedure that reloads the ROM-based image, downloads the software to Flash memory, and reboots to the system image in Flash memory. Flash load helper performs checks and validations to maximize the success of a Flash upgrade and minimize the chance of leaving Flash memory either in an erased state or with a file that cannot boot.

In run-from-Flash systems, the software image is stored in and executed from the Flash EPROM rather than from RAM. This method reduces memory cost. A run-from-Flash system requires enough Flash EPROM to hold the image and enough main system RAM to hold the routing tables and data structures. The system does not need the same amount of main system RAM as a run-from-RAM system because the full image does not reside in RAM. Run-from-Flash systems include the Cisco 2500 series and some Cisco 3000 series.

## Flash Load Helper Features

Flash load helper includes the following features:

- Confirms access to the specified source file on the specified server before erasing Flash memory and reloading to the ROM image for the actual upgrade.
- Warns you if the image being downloaded is not appropriate for the system.
- Prevents reloads to the ROM image for a Flash upgrade if the system is not set up for automatic booting and the user is not on the console terminal. In the event of a catastrophic failure during the upgrade, Flash load helper can bring up the boot ROM image as a last resort rather than forcing the system to wait at the ROM monitor prompt for input from the console terminal.
- Retries Flash downloads automatically up to six times. The retry sequence is as follows:
  - First try
  - Immediate retry
  - Retry after 30 seconds
  - Reload ROM image and retry
  - Immediate retry
  - Retry after 30 seconds
- Allows you to save any configuration changes made before you exit out of the system image.
- Notifies users logged in to the system of the impending switch to the boot ROM image so that they do not lose their connections unexpectedly.
- Logs console output during the Flash load helper operation into a buffer that is preserved through system reloads. You can retrieve the buffer contents from a running image. The output is useful when console access is unavailable or a failure occurs in the download operation.

Flash load helper can also be used on systems with multiple banks of Flash memory that support Flash memory partitioning. Flash load helper enables you to download a new file into the same partition from which the system is executing an image.

For information about how to partition multiple banks of Flash memory so your system can hold two different images, see the “Partitioning Flash Memory” section.

## Flash Load Helper Configuration Task List

Perform the commands in the following sections to use and monitor Flash load helper:

- Download a File Using Flash Load Helper
- Monitor Flash Load Helper

## Download a File Using Flash Load Helper

To download a new file to Flash memory using Flash load helper, check to make sure that your boot ROMs support Flash load helper and then use one of the following commands in privileged EXEC mode:

Command	Purpose
<pre>copy tftp: flash: copy rcp: flash: copy ftp: flash:</pre>	Download a new file to Flash memory.

The following error message displays if you are in a Telnet session and the system is set for manual booting (the boot bits in the configuration register are zero):

```
ERR: Config register boot bits set for manual booting
```

In case of any catastrophic failure in the Flash memory upgrade, this error message helps to minimize the chance of the system going down to ROM monitor mode and being taken out of the remote Telnet user's control.

The system tries to bring up at least the boot ROM image if it cannot boot an image from Flash memory. Before reinitiating the **copy:** command, you must set the configuration register boot field to a nonzero value, using the **config-register** global configuration command.

The **copy** command initiates a series of prompts to which you must provide responses. The dialog is similar to the following:

```
Router# copy tftp: flash:

***** NOTICE *****
Flash load helper v1.0
This process will accept the TFTP copy options and then terminate
the current system image to use the ROM based image for the copy.
Router functionality will not be available during that time. If
you are logged in via telnet, this connection will terminate. Users
with console access can see the results of the copy operation.
*****

There are active users logged into the system.

Proceed? [confirm] y
System flash directory:
File Length  Name/status
1    2251320 abc/igs-kf.914
[2251384 bytes used, 1942920 available, 4194304 total]
Address or name of remote host [255.255.255.255]? 172.16.1.111
Source file name? abc/igs-kf.914
Destination file name [default = source name]? <Return>
Accessing file 'abc/igs-kf.914' on 172.16.1.111...
Loading from 172.16.13.111:
Erase flash device before writing? [confirm] n
File 'abc/igs-kf.914' already exists; it will be invalidated!
Invalidate existing copy of 'abc/igs-kf.914' in flash memory? [confirm] y
Copy 'abc/igs-kf.914' from TFTP server
as 'abc/igs-kf.914' into Flash WITHOUT erase? y

%SYS-5-RELOAD: Reload requested
%FLH: rxboot/igs-kf.914r from 172.16.1.111 to flash...
```

The Flash Load Helper operation verifies the request from the running image by trying to copy a single block from the remote server. Then the Flash load helper is executed, causing the system to reload to the ROM-based system image. If the file does not seem to be a valid image for the system, a warning is displayed and a separate confirmation is sought from you.

If the configuration has been modified but not yet saved, you are prompted to save the configuration:

```
System configuration has been modified. Save? [confirm]
```

Users with open Telnet connections are notified of the system reload, as follows:

```
**System going down for Flash upgrade**
```

If the copy process fails, the copy operation is retried up to three times. If the failure happens in the middle of a copy operation so that only part of the file has been written to Flash memory, the retry does not erase Flash memory unless you specified an erase operation. The partly written file is marked as deleted, and a new file is opened with the same name. If Flash memory runs out of free space in this process, the copy operation is terminated.

After Flash load helper finishes copying (whether the copy operation is successful or not), it automatically attempts an automatic or a manual boot, depending on the value of bit zero of the configuration register boot field according to the following:

- If bit zero equals 0, the system attempts a default boot from Flash memory to load up the first bootable file in Flash memory. This default boot is equivalent to a manual **boot flash** command at the ROM monitor prompt.
- If bit zero equals 1, the system attempts to boot based on the boot configuration commands. If no boot configuration commands exist, the system attempts a default boot from Flash memory; that is, it attempts to load the first bootable file in Flash memory.

## Monitor Flash Load Helper

To view the system console output generated during the Flash load helper operation, use the image that has been booted up after the Flash memory upgrade. Use the following command in privileged EXEC mode:

Command	Purpose
<code>more flh:logfile</code>	View the console output generated during the Flash load helper operation.

If you are a remote Telnet user performing the Flash upgrade without a console connection, this task allows you to retrieve console output when your Telnet connection has terminated due to the switch to the ROM image. The output indicates what happened during the download, and is particularly useful if the download fails.

## Formatting Flash Memory

On Class A and Class C Flash file systems, you can format Flash memory. Formatting erases all information in Flash memory.

On the Cisco 7000 family, you must format a new Flash memory card before using it in a PCMCIA slot.

Flash memory cards have sectors that can fail. You can reserve certain Flash memory sectors as “spares” for use when other sectors fail. Use the **format** command to specify between 0 and 16 sectors as spares. If you reserve a small number of spare sectors for emergencies, you do not waste space because you can use most of the Flash memory card. If you specify zero spare sectors and some sectors fail, you must reformat the Flash memory card and thereby erase all existing data.

The format operation requires at least Cisco IOS Release 11.0 system software.

## Flash Memory Formatting Process



### Caution

The following formatting procedure erases all information in Flash memory. To prevent the loss of important data, proceed carefully.

Use the following procedure to format Flash memory. If you are formatting internal Flash memory, such as bootflash, you can skip the first step. If you are formatting a Flash memory card, complete both steps.

- 
- Step 1** Insert the new Flash memory card into a PCMCIA slot. Refer to instructions on maintaining the router and replacing PCMCIA cards in your router’s hardware documentation for instructions on performing this step.
- Step 2** Format Flash memory.
- 

To format Flash memory, use the following EXEC mode command:

Command	Purpose
<b>format</b> [ <b>spare</b> <i>spare-number</i> ] <i>device1</i> : [[ <i>device2</i> :][ <i>monlib-filename</i> ]] (Class A Flash file systems)	Formats Flash memory.

The following example shows the **format** command that formats a Flash memory card inserted in slot 0.

```
Router# format slot0:
Running config file on this device, proceed? [confirm]y
All sectors will be erased, proceed? [confirm]y
Enter volume id (up to 31 characters): <Return>
Formatting sector 1 (erasing)
Format device slot0 completed
```

When the router returns you to the EXEC prompt, the new Flash memory card is successfully formatted and ready for use.

## Recovering from Locked Blocks

To recover from locked blocks, reformat the Flash memory card. A locked block of Flash memory occurs when power is lost or a Flash memory card is unplugged during a write or erase operation. When a block of Flash memory is locked, it cannot be written to or erased, and the operation will consistently fail at a particular block location. The only way to recover from locked blocks is by reformatting the Flash memory card with the **format** command.

**Caution**

Formatting a Flash memory card to recover from locked blocks will cause existing data to be lost.

## Using Memory Scan on the Cisco 7500 Series

On Cisco 7500 series routers (including 7000 series with the RSP7000 card upgrade), a memory scanning feature is available. This feature adds a low-priority background process that searches all installed dynamic random-access memory (DRAM) for possible parity errors. If errors are found in memory areas that are not in use, this feature attempts to scrub (remove) the errors. The time to complete one memory scan and scrub cycle can range from 10 minutes to several hours, depending on the amount of installed memory. The impact of the Memory Scan feature on the central processing unit (CPU) is minimal. The feature can be controlled and monitored with the new **memory scan** and **show memory scan** command-line interface (CLI) commands.

The Memory Scan feature does not discriminate against different information types in DRAM; that is, it perceives text, data, and heap information in the same way. The feature continues to work when a memory cell is busy, although it might respond differently to errors found in different areas. The feature responds to errors in one or more of the following ways:

- A message is logged for all errors found. Each message contains an explanation of the error and suggests corrective action if applicable.
- For errors in heap storage control blocks, attempts are made to scrub errors in the free blocks. If an error is scrubbed, no further action occurs, but there is an entry in the error log. If it is not scrubbed, the block that contains the error is linked to a bad-memory list which will not be allocated to users. If the memory block is large, the block is split and only a small portion containing the error is linked to a bad-memory list.
- For errors in a busy block, or in other areas such as text or data, an error message is produced but no further action is taken, preventing damage to living data.

## Configuring and Verifying Memory Scan

Use the **memory scan** command in global configuration mode to enable the feature.

Use the **more system:running-configuration** command in privileged EXEC mode to verify that memory scan appears in the running configuration.

Use the **show memory scan** command to monitor the number and type of parity errors on your system. Use the **show memory scan** command in privileged EXEC mode. In the following example, the feature is enabled and no parity errors are found:

```
Router# show memory scan
Memory scan is on.
No parity error has been detected.
```

If the Memory Scan feature has not been configured, or has been turned off, the **show memory scan** command generates a report. In the following example, memory scan is turned off:

```
Router# show memory scan
Memory scan is off
No parity error has been detected.
```

If errors are detected in the system, the **show memory scan** command generates an error report. In the following example, memory scan detected a parity error:

```
Router# show memory scan
Memory scan is on.
Total Parity Errors 1.
Address BlockPtr BlckSize Disposit Region Timestamp
6115ABCD 60D5D090 9517A4 Scrubed Local 16:57:09 UTC Thu Mar 18
```

For an explanation of the error report fields, see the full details on the **show memory scan** command in the “Router Memory Commands” chapter of the Release 12.1 *Cisco IOS Configuration Fundamentals Command Reference*.