# Creating Core Dumps

When a router crashes, it is sometimes useful to obtain a full copy of the memory image (called a *core dump*) to identify the cause of the crash. Core dumps are generally only useful to your technical support representative.

**Caution** Use the commands discussed in this appendix only under the direction of a technical support representative. Creating a core dump while the router is functioning in a network can disrupt network operation. The resulting binary file, which is very large, must be transferred to a TFTP, FTP, or rcp server and subsequently interpreted by technical personnel who have access to source code and detailed memory maps.

This chapter consists of the following sections:

- **exception** Commands
- **write core** Command
- **show** Commands

## exception Commands

Use the **exception** class of configuration commands only after consulting with a technical support representative. These commands are useful for debugging purposes, but they can result in unexpected behavior.

### Creating a Core Dump

To obtain a core dump when a router crashes, use the **exception dump** *ip-address* router configuration command (where *ip-address* is the address of your TFTP server).

Including this command in your configuration causes the router to attempt to make a core dump when it crashes. The core dump is written to a file named *hostname*-core on your TFTP server, where *hostname* is the name of the router. You can change the name of the core file by configuring the **exception core-file** *filename* command.

This procedure can fail for certain types of system crashes. However, if successful, the core dump file will be the size of the memory available on the processor (for example, 16 MB for a CSC/4).

### Creating an Exception Memory Core Dump

During the debugging process, you can cause the router to create a core dump and reboot when certain memory size parameters are violated. The **exception memory** commands define a minimum contiguous block of memory in the free pool and a minimum size for the free memory pool.

[**no**] **exception memory fragment** *size*

[**no**] **exception memory minimum** *size*

The value of *size* is in bytes and is checked every 60 seconds. If you enter a size that is greater than the free memory and the **exception dump** command has been configured, a core dump and router reload is generated after 60 seconds. If the **exception dump** command is not configured, the router reloads without generating a core dump.

The following example configures the router to monitor the free memory. If it falls below 250,000 bytes, it will dump the core and reload.

```
exception dump 131.108.92.2
exception core-file memory.overrun
exception memory minimum 250000
```

## write core Command

You can test core dumps by using the **write core** privileged EXEC command. This command causes the router to generate a core dump without reloading and is useful if the router is malfunctioning but has not crashed.

Depending on your TFTP server, you might need to create an empty target file to which the router can write the core.

## show Commands

When a router fails with an unexpected reload and you report the problem to a technical support representative, always include a copy of the output from the **show stacks** and **show version** EXEC commands. Output from these commands provides the support representative with important information about the state of your router when it failed.

### show stacks Command

The **show stacks** command displays data saved by the ROM monitor, which includes a failure type, an operand address, and a failure program counter. This data is overwritten when the system is reloaded, so check your configuration register settings and decide how you want to recover from system crashes.

The “Memory Maps” appendix provides an example of **show stacks** output and memory map information that can help you determine whether a system crash was caused by a software or hardware problem.

## show version Command

The **show version** command displays the image type, version number, and function sets that identify the exact software that is running on your router. Also displayed is the current configuration register setting.

```
milou>show version
Cisco Internetwork Operating System Software
IOS (tm) GS Software (GS7-K-M), Version 11.0(9), RELEASE SOFTWARE (fc1)
Copyright (c) 1986-1996 by cisco Systems, Inc.
Compiled Tue 11-Jun-96 03:52 by tstevens
Image text-base: 0x00001000, data-base: 0x007614F0

ROM: System Bootstrap, Version 5.2(2), RELEASE SOFTWARE
ROM: GS Software (GS7), Version 10.2(2), RELEASE SOFTWARE (fc1)

milou uptime is 2 days, 20 hours, 26 minutes
System restarted by error - Software forced crash, PC 0x1CF82C at 20:25:38 PDT M
on Aug 5 1996
System image file is “images/gs7-k-mz.110-9”, booted via flash

cisco RP1 (68040) processor (revision B0) with 16384K bytes of memory.
Processor board ID 00130334
G.703/E1 software, Version 1.0.
Bridging software.
X.25 software, Version 2.0, NET2, BFE and GOSIP compliant.
Primary Rate ISDN software, Version 1.0.
1 Silicon Switch Processor.
1 EIP controller (6 Ethernet).
1 FEIP controller (2 FastEthernet).
1 FIP controller (1 FDDI).
1 MIP controller (2 T1)
6 Ethernet/IEEE 802.3 interfaces.
2 FastEthernet/IEEE 802.3 interfaces.
48 Serial network interfaces.
1 FDDI network interface.
2 Channelized T1/PRI ports.
128K bytes of non-volatile configuration memory.
4096K bytes of flash memory sized on embedded flash.

Configuration register is 0x102
milou>
```

### Version Numbering

Cisco uses a numbering scheme that a uniquely identifies each release of the Cisco IOS software. Understanding this scheme will help you distinguish between the different types of releases that are available.

The following formula is used to identify releases of the Cisco IOS software:

**A.a (x.y)**

Major releases are indicated by the numbers outside of the parentheses (*A* and *a*). Examples of major release numbers are 9.21, 10.0, 10.2, 11.1, and so on. The initial release available for customers is indicated by a “1” in parentheses following the major release number, for example, 9.21(1), 10.0(1), 10.2(1), and 11.1(1).

Maintenance releases are indicated by a whole number (*x*) within the parentheses. Each periodic maintenance release number is incremented sequentially, for example, 10.2(2), 10.2(3), 10.2(4), 10.2(5), and so on.

Be extremely cautious with any release that has a number following the maintenance release number inside the parentheses (*y*), for example, the ".5" in parentheses in the version number 10.2(3.5). These numbers indicate that this version of software is an interim build. Interim builds are unit tested, but have *not* been fully regression tested and should be used only for short-term, urgent point-fix situations until the next maintenance release is available.