



Configuring the Router for the First Time

This chapter contains information about how to initially configure the Cisco 7600 series router. The information in this chapter supplements the administration information and procedures in these publications:

- *Cisco IOS Configuration Fundamentals Configuration Guide*, Release 12.2, at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/ffun_c.html
- *Cisco IOS Configuration Fundamentals Configuration Command Reference*, Release 12.2, at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/configfun/command/reference/ffun_r.html



Note

For complete syntax and usage information for the commands used in this chapter, refer to these publications:

- The *Cisco 7600 Series Router Cisco IOS Command Reference* at this URL:
http://www.cisco.com/en/US/products/hw/routers/ps368/prod_command_reference_list.html

This chapter consists of these sections:

- [Default Configuration](#), page 2-1
- [Configuring the Router](#), page 2-2
- [Protecting Access to Privileged EXEC Commands](#), page 2-8
- [Recovering a Lost Enable Password](#), page 2-13
- [Modifying the Supervisor Engine Startup Configuration](#), page 2-13

Default Configuration

[Table 2-1](#) shows the default configuration.

Table 2-1 **Default Configuration**

Feature	Default Value
Administrative connection	Normal mode
Global information	No value for the following: <ul style="list-style-type: none">• System name• System contact• Location
System clock	No value for system clock time
Passwords	No passwords configured for normal mode or enable mode (press the Return key)
Prompt	Router>

Configuring the Router

These sections describe how to configure the router:

- [Using the Setup Facility or the setup Command, page 2-2](#)
- [Using Configuration Mode, page 2-3](#)
- [Checking the Running Configuration Before Saving, page 2-4](#)
- [Saving the Running Configuration Settings, page 2-5](#)
- [Reviewing the Configuration, page 2-5](#)
- [Configuring a Static Route, page 2-5](#)
- [Configuring a Static Route, page 2-5](#)
- [Configuring a BOOTP Server, page 2-6](#)

Using the Setup Facility or the setup Command

At initial startup, the router automatically defaults to the setup facility. You can also invoke the setup facility by entering the **setup** command at the enable prompt (#).

The setup facility provides a System Configuration Dialog, which is an interactive CLI mode that guides you through first-time configuration of the router. The dialog prompts you for the information needed to start your router functioning in the network.

The System Configuration Dialog first prompts you to configure global parameters, which are used to control system-wide settings. The dialog then prompts for information to configure interfaces. You must progress through the System Configuration Dialog until you reach an item you want to change.

As you move through the dialog, square brackets beside each prompt show the default setting for that item or the last configured value. To accept the default value for an item, press **Return** or **Enter**. To change the value for that item, enter the desired value.

To display help for a prompt, press the question mark (?) key at the prompt.

When you complete your changes, the system automatically displays the configuration file that was created during the setup session. The dialog asks if you want to use this configuration. If you answer Yes, the configuration is saved to NVRAM as the startup configuration file. If you answer No, the configuration is not saved and the process begins again.

To exit setup and return to privileged EXEC mode without making changes and without progressing through the entire dialog, press **Ctrl-C**.

When you complete the configuration process, your interfaces are now available for limited use. If you want to modify the currently saved configuration parameters after the initial configuration, enter the **setup** command. To perform more complex configurations, enter configuration mode and use the **configure** command.

**Note**

You can use the **show version** command to check the current state of the router.

For detailed interface configuration information, refer to the *Cisco IOS Interface Configuration Guide* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/finter_c/index.htm

Using Configuration Mode

If you prefer not to use the setup facility, you can configure the router from configuration mode as follows:

- Step 1** Connect a console terminal to the console interface of your supervisor engine.
- Step 2** When you are asked if you want to enter the initial dialog, answer **no** to enter the normal operating mode as follows:

```
Would you like to enter the initial dialog? [yes]: no
```

- Step 3** After a few seconds you will see the user EXEC prompt (**Router>**). Type **enable** to enter enable mode:

```
Router> enable
```

**Note**

Configuration changes can only be made in enable mode.

The prompt will change to the privileged EXEC prompt (**#**) as follows:

```
Router#
```

- Step 4** At the prompt (**#**), enter the **configure terminal** command to enter configuration mode as follows:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

At the prompt, enter the **interface type slot/interface** command to enter interface configuration mode as follows:

```
Router(config)# interface fastethernet 5/1
Router(config-if)#
```

In either of these configuration modes, you can enter any changes to the configuration. Enter the **end** command to exit configuration mode.

Step 5 Save your settings. (See the “[Saving the Running Configuration Settings](#)” section on page 2-5.)

Your router is now minimally configured and can boot with the configuration you entered. To see a list of the configuration commands, enter **?** at the prompt or press the **help** key in configuration mode.

Checking the Running Configuration Before Saving

You can check the configuration settings you entered or changes you made by entering the **show running-config** command at the privileged EXEC prompt (#) as follows:

```
Router# show running-config
Building configuration...

Current configuration:
Current configuration : 3441 bytes
!
version 12.1
service timestamps debug datetime localtime
service timestamps log datetime localtime
no service password-encryption
!
hostname Router
!
boot buffersize 522200
boot system flash disk0:c6sup22-jsv-mz.121-5c.EX.bin
enable password lab
!
redundancy
  main-cpu
    auto-sync standard
ip subnet-zero
no ip finger
!
cns event-service server
!
<...output truncated...>
!
interface FastEthernet3/3
  ip address 172.20.52.19 255.255.255.224
!
<...output truncated...>
!
line con 0
  exec-timeout 0 0
  transport input none
line vty 0 4
  exec-timeout 0 0
  password lab
  login
  transport input lat pad mop telnet rlogin udptn nasi
!
end
Router#
```

Saving the Running Configuration Settings

To store the configuration or changes to your startup configuration in NVRAM, enter the **copy running-config startup-config** command at the privileged EXEC prompt (#) as follows:

```
Router# copy running-config startup-config
```

This command saves the configuration settings that you created in configuration mode. If you fail to do this step, your configuration will be lost the next time you reload the system.

Reviewing the Configuration

To display information stored in NVRAM, enter the **show startup-config** EXEC command. The display should be similar to the display from the **show running-config** EXEC command.

Configuring a Static Route

If your Telnet station or SNMP network management workstation is on a different network from your router and a routing protocol has not been configured, you might need to add a static routing table entry for the network where your end station is located.

To configure a static route, perform this task:

	Command	Purpose
Step 1	Router(config)# ip route <i>dest_IP_address mask</i> { <i>forwarding_IP</i> vlan <i>vlan_ID</i> }	Configures a static route.
Step 2	Router# show running-config	Verifies the static route configuration.

This example shows how to use the **ip route** command to configure a static route to a workstation at IP address 171.10.5.10 on the router with a subnet mask and IP address 172.20.3.35 of the forwarding router:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# ip route 171.10.5.10 255.255.255.255 172.20.3.35
Router(config)# end
Router#
```

This example shows how to use the **show running-config** command to confirm the configuration of the previously configured static route:

```
Router# show running-config
Building configuration...
.
<...output truncated...>
.
ip default-gateway 172.20.52.35
ip classless
ip route 171.10.5.10 255.255.255.255 172.20.3.35
no ip http server
!
line con 0
  transport input none
line vty 0 4
```

```

exec-timeout 0 0
password lab
login
transport input lat pad dsipcon mop telnet rlogin udptn nasi
!
end

```

Router#

This example shows how to use the **ip route** command to configure a static route to a workstation at IP address 171.20.5.3 on the router with subnet mask and connected over VLAN 1:

```

Router# configure terminal
Router(config)# ip route 171.20.5.3 255.255.255.255 vlan 1
Router(config)# end
Router#

```

This example shows how to use the **show running-config** command to confirm the configuration of the previously configured static route:

```

Router# show running-config
Building configuration...
.
<...output truncated...>
.
ip default-gateway 172.20.52.35
ip classless
ip route 171.20.52.3 255.255.255.255 Vlan1
no ip http server
!
!
x25 host z
!
line con 0
  transport input none
line vty 0 4
  exec-timeout 0 0
  password lab
  login
  transport input lat pad dsipcon mop telnet rlogin udptn nasi
!
end

Router#

```

Configuring a BOOTP Server

The Bootstrap Protocol (BOOTP) automatically assigns an IP address by adding the MAC and IP addresses of the interface to the BOOTP server configuration file. When the router boots, it automatically retrieves the IP address from the BOOTP server.

The router performs a BOOTP request *only* if the current IP address is set to 0.0.0.0. (This address is the default address for a new router or a router that has had its startup-config file cleared using the **erase** command.)

To allow your router to retrieve its IP address from a BOOTP server, you must first determine the MAC address of the router and add that MAC address to the BOOTP configuration file on the BOOTP server. To create a BOOTP server configuration file, follow these steps:

-
- Step 1** Install the BOOTP server code on the workstation, if it is not already installed.

- Step 2** Determine the MAC address from the label on the chassis.
- Step 3** Add an entry in the BOOTP configuration file (usually /usr/etc/bootptab) for each router. Press **Return** after each entry to create a blank line between each entry. See the example BOOTP configuration file that follows in Step 4.
- Step 4** Enter the **reload** command to reboot and automatically request the IP address from the BOOTP server.

This example BOOTP configuration file shows the added entry:

```
# /etc/bootptab: database for bootp server (/etc/bootpd)
#
# Blank lines and lines beginning with '#' are ignored.
#
# Legend:
#
#      first field -- hostname
#                      (may be full domain name and probably should be)
#
#      hd -- home directory
#      bf -- bootfile
#      cs -- cookie servers
#      ds -- domain name servers
#      gw -- gateways
#      ha -- hardware address
#      ht -- hardware type
#      im -- impress servers
#      ip -- host IP address
#      lg -- log servers
#      lp -- LPR servers
#      ns -- IEN-116 name servers
#      rl -- resource location protocol servers
#      sm -- subnet mask
#      tc -- template host (points to similar host entry)
#      to -- time offset (seconds)
#      ts -- time servers
#
<information deleted>
#
#####
# Start of individual host entries
#####
Router:      tc=netcisco0:   ha=0000.0ca7.ce00:      ip=172.31.7.97:
dross:      tc=netcisco0:   ha=00000c000139:      ip=172.31.7.26:
<information deleted>
```

Route Health Injection on Cisco 7600

Route Health Injection (RHI) allows the Cisco Application Control Engine (ACE) to advertise the availability of a virtual IP (VIP) address throughout the intranet as a host route. The ACE sends the RHI information to the Multilayer Switch Feature Card (MSFC) in the Cisco 7600 series router, which periodically propagates the VIP availability according to the RHI information it receives. RHI is normally restricted to intranets because the MSFC does not broadcast host-route availability to the Internet.

When a VIP becomes unavailable, the ACE withdraws the RHI information. The MSFC adds an entry in its routing table for each VIP address it receives from the ACE. The routing protocol running on the MSFC sends routing-table updates, including availability and hop-count routing information for each

instance of a VIP address to other routers. The client router uses the routing information to choose a route based on the best available path to that VIP address, and also where the Cisco application switch is logically closer to the client system.

By default, the ACE advertises the VLAN of the VIP interface for RHI. To advertise a VLAN for route health injection (RHI) that is different from the VIP interface VLAN, use the **ip route inject vlan** command in interface configuration mode. By default, the ACE advertises the VLAN of the VIP interface for RHI.

Effective with Cisco IOS Release 15.1(3)S, Cisco 7600 routers support IPv6 static route injection.

For the detailed configuration and examples, see the following guides:

Routing and Bridging Guide vA2(1.0), Cisco ACE Application Control Engine Module at

http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A2/configuration/rig_brdg/guide/rigbrdg.html

Getting Started Guide, Cisco ACE Application Control Engine Module at

http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/vA5_1_0/configuration/getting_started/guide/ace_module_gsg.html

Protecting Access to Privileged EXEC Commands

The following tasks provide a way to control access to the system configuration file and privileged EXEC commands:

- [Setting or Changing a Static Enable Password, page 2-8](#)
- [Using the enable password and enable secret Commands, page 2-9](#)
- [Setting or Changing a Line Password, page 2-9](#)
- [Setting TACACS+ Password Protection for Privileged EXEC Mode, page 2-10](#)
- [Encrypting Passwords, page 2-10](#)
- [Configuring Multiple Privilege Levels, page 2-11](#)

Setting or Changing a Static Enable Password

To set or change a static password that controls access to the privileged EXEC mode, perform this task:

Command	Purpose
Router(config)# enable password <i>password</i>	Sets a new password or changes an existing password for the privileged EXEC mode.

This example shows how to configure an enable password as “lab” at the privileged EXEC mode:

```
Router# configure terminal
Router(config)# enable password lab
Router(config)#
```

To display the password or access level configuration, see the “[Displaying the Password, Access Level, and Privilege Level Configuration](#)” section on page 2-12.

Using the enable password and enable secret Commands

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a TFTP server, you can use either the **enable password** or **enable secret** commands. Both commands configure an encrypted password that you must enter to access enable mode (the default) or to access a specified privilege level. We recommend that you use the **enable secret** command.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

To configure the router to require an enable password, perform either of these tasks:

Command	Purpose
Router(config)# enable password [level <i>level</i>] { <i>password</i> <i>encryption-type</i> <i>encrypted-password</i> }	Establishes a password for the privileged EXEC mode.
Router(config)# enable secret [level <i>level</i>] { <i>password</i> <i>encryption-type</i> <i>encrypted-password</i> }	Specifies a secret password, saved using a nonreversible encryption method. (If enable password and enable secret commands are both set, users must enter the enable secret password.)

Use either of these commands with the **level** option to define a password for a specific privilege level. After you specify the level and set a password, give the password only to users who need to have access at this level. Use the **privilege level** configuration command to specify commands accessible at various levels.

If you enable the **service password-encryption** command, the password you enter is encrypted. When you display it with the **more system:running-config** command, it displays in encrypted form.

If you specify an encryption type, you must provide an encrypted password that you copy from another Cisco 7600 series router configuration.

**Note**

You cannot recover a lost encrypted password. You must clear NVRAM and set a new password. See the [“Recovering a Lost Enable Password”](#) section on page 2-13 if you lose or forget your password.

To display the password or access level configuration, see the [“Displaying the Password, Access Level, and Privilege Level Configuration”](#) section on page 2-12.

Setting or Changing a Line Password

To set or change a password on a line, perform this task:

Command	Purpose
Router(config-line)# password <i>password</i>	Sets a new password or change an existing password for the privileged level.

To display the password or access level configuration, see the [“Displaying the Password, Access Level, and Privilege Level Configuration”](#) section on page 2-12.

Setting TACACS+ Password Protection for Privileged EXEC Mode

For complete information about TACACS+, refer to these publications:

- *Cisco IOS Security Configuration Guide*, Release 12.2, “Authentication, Authorization, and Accounting (AAA),” at this URL:

http://www.cisco.com/en/US/docs/ios/12_2/security/command/reference/fsecur_r.html

- *Cisco IOS Security Command Reference*, Release 12.2, at this URL:

http://www.cisco.com/en/US/docs/ios/12_2/security/command/reference/fsecur_r.html

To set the TACACS+ protocol to determine whether or not a user can access privileged EXEC mode, perform this task:

Command	Purpose
Router(config)# enable use-tacacs	Sets the TACACS-style user ID and password-checking mechanism for the privileged EXEC mode.

When you set TACACS password protection at the privileged EXEC mode, the **enable** EXEC command prompts for both a new username and a password. This information is then sent to the TACACS+ server for authentication. If you are using the extended TACACS+, it also sends any existing UNIX user identification code to the TACACS+ server.



Caution

If you enter the **enable use-tacacs** command, you must also enter **tacacs-server authenticate enable**, or you are locked out of the privileged EXEC mode.



Note

When used without extended TACACS, the **enable use-tacacs** command allows anyone with a valid username and password to access the privileged EXEC mode, creating a potential security problem. This problem occurs because the router cannot tell the difference between a query resulting from entering the **enable** command and an attempt to log in without extended TACACS.



Note

Do not use the standby supervisor port to connect to the AAA-server or for any traffic and authentication related functionalities. The router may become inaccessible for 5 minutes after reload if the standby supervisor port is used.

Encrypting Passwords

Because protocol analyzers can examine packets (and read passwords), you can increase access security by configuring the Cisco IOS software to encrypt passwords. Encryption prevents the password from being readable in the configuration file.

To configure the Cisco IOS software to encrypt passwords, perform this task:

Command	Purpose
Router(config)# service password-encryption	Encrypts a password.

Encryption occurs when the current configuration is written or when a password is configured. Password encryption is applied to all passwords, including authentication key passwords, the privileged command password, console and virtual terminal line access passwords, and Border Gateway Protocol (BGP) neighbor passwords. The **service password-encryption** command keeps unauthorized individuals from viewing your password in your configuration file.

**Caution**

The **service password-encryption** command does not provide a high level of network security. If you use this command, you should also take additional network security measures.

Although you cannot recover a lost encrypted password (that is, you cannot get the original password back), you can regain control of the router after you lose or forget the encrypted password. See the [“Recovering a Lost Enable Password” section on page 2-13](#) if you lose or forget your password.

To display the password or access level configuration, see the [“Displaying the Password, Access Level, and Privilege Level Configuration” section on page 2-12](#).

Configuring Multiple Privilege Levels

By default, the Cisco IOS software has two modes of password security: user EXEC mode and privileged EXEC mode. You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password widely. If you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to more restricted users.

These tasks describe how to configure additional levels of security:

- [Setting the Privilege Level for a Command, page 2-11](#)
- [Changing the Default Privilege Level for Lines, page 2-12](#)
- [Logging In to a Privilege Level, page 2-12](#)
- [Exiting a Privilege Level, page 2-12](#)
- [Displaying the Password, Access Level, and Privilege Level Configuration, page 2-12](#)

Setting the Privilege Level for a Command

To set the privilege level for a command, perform this task:

	Command	Purpose
Step 1	Router(config)# privilege mode level level <i>command</i>	Sets the privilege level for a command.
Step 2	Router(config)# enable password level level <i>[encryption-type] password</i>	Specifies the enable password for a privilege level.

To display the password or access level configuration, see the [“Displaying the Password, Access Level, and Privilege Level Configuration”](#) section on page 2-12.

Changing the Default Privilege Level for Lines

To change the default privilege level for a given line or a group of lines, perform this task:

Command	Purpose
Router(config-line)# privilege level level	Changes the default privilege level for the line.

To display the password or access level configuration, see the [“Displaying the Password, Access Level, and Privilege Level Configuration”](#) section on page 2-12.

Logging In to a Privilege Level

To log in at a specified privilege level, perform this task:

Command	Purpose
Router# enable level	Logs into a specified privilege level.

Exiting a Privilege Level

To exit to a specified privilege level, perform this task:

Command	Purpose
Router# disable level	Exits to a specified privilege level.

Displaying the Password, Access Level, and Privilege Level Configuration

To display the password, access level, and privilege level configuration, perform this task:

	Command	Purpose
Step 1	Router# show running-config	Displays the password and the access level configuration.
Step 2	Router# show privilege	Shows the privilege level configuration.

This example shows how to display the password and access level configuration:

```
Router# show running-config
<...output truncated...>
enable password lab
<...output truncated...>
```

This example shows how to display the privilege level configuration:

```
Router# show privilege
Current privilege level is 15
Router#
```

Recovering a Lost Enable Password

To recover a lost enable password, follow these steps:

-
- | | |
|---------------|---|
| Step 1 | Connect to the console interface. |
| Step 2 | Configure the router to boot up without reading the configuration memory (NVRAM). |
| Step 3 | Reboot the system. |
| Step 4 | Access enable mode (which can be done without a password when one is not configured). |
| Step 5 | View or change the password, or erase the configuration. |
| Step 6 | Reconfigure the router to boot up and read the NVRAM as it normally does. |
| Step 7 | Reboot the system. |
-

**Note**

Password recovery requires the Break signal. You must be familiar with how your terminal or PC terminal emulator issues this signal. For example, in ProComm, the Alt-B keys generate the Break signal. In a Windows terminal session, you press the **Break** or **Ctrl** and **Break** keys simultaneously.

Modifying the Supervisor Engine Startup Configuration

These sections describe how the startup configuration on the supervisor engine works and how to modify the configuration register and BOOT variable:

- [Understanding the Supervisor Engine Boot Configuration, page 2-13](#)
- [Configuring the Software Configuration Register, page 2-14](#)
- [Specifying the Startup System Image, page 2-17](#)
- [Understanding Flash Memory, page 2-18](#)
- [CONFIG_FILE Environment Variable, page 2-19](#)
- [Controlling Environment Variables, page 2-19](#)

Understanding the Supervisor Engine Boot Configuration

These next sections describe how the boot configuration works on the supervisor engine.

Understanding the Supervisor Engine Boot Process

The supervisor engine boot process involves two software images: ROM monitor and supervisor engine software. When the router is powered up or reset, the ROM-monitor code is executed. Depending on the NVRAM configuration, the supervisor engine either stays in ROM-monitor mode or loads the supervisor engine software.

Two user-configurable parameters determine how the router boots: the configuration register and the BOOT environment variable. The configuration register is described in the [“Modifying the Boot Field and Using the boot Command” section on page 2-15](#). The BOOT environment variable is described in the [“Specifying the Startup System Image” section on page 2-17](#).

Understanding the ROM Monitor

The ROM monitor executes upon power-up, reset, or when a fatal exception occurs. The router enters ROM-monitor mode if the router does not find a valid software image, if the NVRAM configuration is corrupted, or if the configuration register is set to enter ROM-monitor mode. From ROM-monitor mode, you can manually load a software image from bootflash or a Flash PC card.



Note

For complete syntax and usage information for the ROM monitor commands, refer to the *Cisco 7600 Series Router Cisco IOS Command Reference* publication.

You can also enter ROM-monitor mode by restarting and then pressing the **Break** key during the first 60 seconds of startup. If you are connected through a terminal server, you can escape to the Telnet prompt and enter the **send break** command to enter ROM-monitor mode.



Note

The **Break** key is always enabled for 60 seconds after rebooting, regardless of whether the configuration-register setting has the **Break** key disabled.

The ROM monitor has these features:

- Power-on confidence test
- Hardware initialization
- Boot capability (manual boot and autoboot)
- Debug utility and crash analysis
- Monitor call interface (EMT calls—the ROM monitor provides information and some functionality to the running software images through EMT calls)
- File system (the ROM monitor knows the simple file system and supports the newly developed file system through the dynamic linked file system library [MONLIB])
- Exception handling

Configuring the Software Configuration Register

The router uses a 16-bit software configuration register, which allows you to set specific system parameters. Settings for the software configuration register are written into NVRAM. Following are some reasons for changing the software configuration register settings:

- To select a boot source and default boot filename.
- To enable or disable the Break function.
- To control broadcast addresses.
- To set the console terminal baud rate.
- To load operating software from flash memory.
- To recover a lost password.

- To allow you to manually boot the system using the **boot** command at the bootstrap program prompt.
- To force an automatic boot from the system bootstrap software (boot image) or from a default system image in onboard flash memory, and read any **boot system** commands that are stored in the configuration file in NVRAM.

Table 2-2 lists the meaning of each of the software configuration memory bits, and Table 2-3 defines the boot field.

**Caution**

The recommended configuration register setting is 0x2102 (this is the factory default value). If you configure a setting that leaves break enabled and you send a break sequence over a console connection, the router drops into ROMMON.

Table 2-2 Software Configuration Register Bit Meaning

Bit Number	Hexadecimal	Meaning
00 to 03	0x0000 to 0x000F	Boot field (see Table 2-3)
06	0x0040	Causes system software to ignore NVRAM contents
07	0x0080	OEM ¹ bit enabled
08	0x0100	Break disabled
09	0x0200	Use secondary bootstrap
10	0x0400	Internet Protocol (IP) broadcast with all zeros
11 to 12	0x0800 to 0x1000	Console line speed (default is 9600 baud)
13	0x2000	Boot default flash software if network boot fails
14	0x4000	IP broadcasts do not have network numbers
15	0x8000	Enable diagnostic messages and ignore NVRAM contents

1. OEM = original equipment manufacturer.

Table 2-3 Explanation of Boot Field (Configuration Register Bits 00 to 03)

Boot Field	Meaning
00	Stays at the system bootstrap prompt
01	Boots the first system image in onboard flash memory
02 to 0F	Specifies a default filename for booting over the network; enables boot system commands that override the default filename

Modifying the Boot Field and Using the boot Command

The configuration register boot field determines whether or not the router loads an operating system image, and if so, where it obtains this system image. The following sections describe using and setting the configuration register boot field, and the tasks you must perform to modify the configuration register boot field.

Bits 0 through 3 of the software configuration register form the boot field.

**Note**

The factory default configuration register setting for systems and spares is 0x2102.

When the boot field is set to either 0 or 1 (0-0-0-0 or 0-0-0-1), the system ignores any boot instructions in the system configuration file and the following occurs:

- When the boot field is set to 0, you must boot the operating system manually by entering the **boot** command to the system bootstrap program or ROM monitor.
- When the boot field is set to 1, the system boots the first image in the onboard bootflash single in-line memory module (SIMM).
- When the entire boot field equals a value between 0-0-1-0 and 1-1-1-1, the router loads the system image specified by **boot system** commands in the startup configuration file.

You can enter the **boot** command only, or enter the command and include additional boot instructions, such as the name of a file stored in flash memory, or a file that you specify for booting from a network server. If you use the **boot** command without specifying a file or any other boot instructions, the system boots from the default flash image (the first image in onboard flash memory). Otherwise, you can instruct the system to boot from a specific flash image (using the **boot system flash filename** command).

You can also use the **boot** command to boot images stored in the Flash PC cards located in Flash PC card slot 0 or slot 1 on the supervisor engine. If you set the boot field to any bit pattern other than 0 or 1, the system uses the resulting number to form a filename for booting over the network.

You must set the boot field for the boot functions you require.

Modifying the Boot Field

You modify the boot field from the software configuration register. To modify the software configuration register boot field, perform this task:

	Command	Purpose
Step 1	Router# show version	Determines the current configuration register setting.
Step 2	Router# configure terminal	Enters configuration mode, selecting the terminal option.
Step 3	Router(config)# config-register value	Modifies the existing configuration register setting to reflect the way in which you want the router to load a system image.
Step 4	Router(config)# end	Exits configuration mode.
Step 5	Router# reload	Reboots to make your changes take effect.

To modify the configuration register while the router is running Cisco IOS software, follow these steps:

- Step 1** Enter the **enable** command and your password to enter privileged level as follows:

```
Router> enable
Password:
Router#
```

- Step 2** Enter the **configure terminal** command at the EXEC mode prompt (#) as follows:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

- Step 3** Configure the configuration register to 0x2102 as follows:

```
Router(config)# config-register 0x2102
```


Set the contents of the configuration register by entering the **config-register** *value* configuration command, where *value* is a hexadecimal number preceded by 0x (see [Table 2-2 on page 2-15](#)).

Step 4 Enter the **end** command to exit configuration mode. The new value settings are saved to memory; however, the new settings do not take effect until the system software is reloaded by rebooting the system.

Step 5 Enter the **show version EXEC** command to display the configuration register value currently in effect and that will be used at the next reload. The value is displayed on the last line of the screen display, as in this example:

```
Configuration register is 0x141 (will be 0x2102 at next reload)
```

Step 6 Save your settings.

See the “[Saving the Running Configuration Settings](#)” section on page 2-5. However, note that configuration register changes take effect only after the system reloads, such as when you enter a **reload** command from the console.

Step 7 Reboot the system.

The new configuration register value takes effect with the next system boot.

Verifying the Configuration Register Setting

Enter the **show version EXEC** command to verify the current configuration register setting. In ROM-monitor mode, enter the **o** command to verify the value of the configuration register boot field.

To verify the configuration register setting, perform this task:

Command	Purpose
Router# show version include Configuration register	Displays the configuration register setting.

In this example, the **show version** command indicates that the current configuration register is set so that the router does not automatically load an operating system image. Instead, it enters ROM-monitor mode and waits for user-entered ROM monitor commands. The new setting instructs the router to load a system image from commands in the startup configuration file or from a default system image stored on a network server.

```
Router1# show version | include Configuration register
Configuration register is 0x2102
Router#
```

Specifying the Startup System Image

You can enter multiple boot commands in the startup configuration file or in the BOOT environment variable to provide backup methods for loading a system image.



Note

- Store the system software image in the **sup-bootflash:**, **disk0:**, or **disk1:** device (only the Route Switch Processor 720 and Supervisor Engine 720 have **disk1:**).

- Do not store the system software image in the **bootflash:** device, which is on the MSFC and is not accessible at boot time.

The BOOT environment variable is also described in the “Specifying the Startup System Image in the Configuration File” section in the “Loading and Maintaining System Images” chapter of the *Cisco IOS Configuration Fundamentals Configuration Guide*.

Understanding Flash Memory

The following sections describe flash memory:

- [Flash Memory Features, page 2-18](#)
- [Security Features, page 2-18](#)
- [Flash Memory Configuration Process, page 2-18](#)

**Note**

The descriptions in the following sections applies to both the bootflash device and to removable flash memory cards.

Flash Memory Features

The flash memory components allow you to do the following:

- Copy the system image to flash memory using TFTP.
- Copy the system image to flash memory using rcp.
- Boot the system from flash memory either automatically or manually.
- Copy the flash memory image to a network server using TFTP or rcp.
- Boot manually or automatically from a system software image stored in flash memory.

Security Features

The flash memory components support the following security features:

- Flash memory cards contain a write-protect switch that you can use to protect data. You must set the switch to *unprotected* to write data to the Flash PC card.
- The system image stored in flash memory can be changed only from privileged EXEC level on the console terminal.

Flash Memory Configuration Process

To configure your router to boot from flash memory, follow these steps:

-
- Step 1** Copy a system image to flash memory using TFTP or rcp (refer to the *Cisco IOS Configuration Fundamentals Configuration Guide*, Release 12.2, “Cisco IOS File Management,” “Loading and Maintaining System Images,” at this URL:

http://www.cisco.com/en/US/docs/ios/12_2/configfun/command/reference/ffun_r.html

- Step 2** Configure the system to boot automatically from the file in flash memory. You might need to change the configuration register value. See the “[Modifying the Boot Field and Using the boot Command](#)” section on [page 2-15](#), for more information on modifying the configuration register.
- Step 3** Save your configurations.
- Step 4** Power cycle and reboot your system to ensure that all is working as expected.
-

CONFIG_FILE Environment Variable

For class A flash file systems, the CONFIG_FILE environment variable specifies the file system and filename of the configuration file to use for initialization (startup). Valid file systems can include **nvram:**, **disk0:**, and **sup-bootflash:**.

For detailed file management configuration information, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/ffun_c/index.htm

After you save the CONFIG_FILE environment variable to your startup configuration, the router checks the variable upon startup to determine the location and filename of the configuration file to use for initialization.

The router uses the NVRAM configuration during initialization when the CONFIG_FILE environment variable does not exist or when it is null (such as at first-time startup). If the router detects a problem with NVRAM or a checksum error, the router enters **setup** mode. See the “[Using the Setup Facility or the setup Command](#)” section on [page 2-2](#) for more information on the **setup** command facility.

Controlling Environment Variables

Although the ROM monitor controls environment variables, you can create, modify, or view them with certain commands. To create or modify the BOOT and CONFIG_FILE environment variables, use the **boot system** and **boot config** global configuration commands.

Refer to the “Specifying the Startup System Image in the Configuration File” section in the “Loading and Maintaining System Images” chapter of the *Configuration Fundamentals Configuration Guide* for details on setting the BOOT environment variable. Refer to the “Specifying the CONFIG_FILE Environment Variable on Class A Flash File Systems” section in the “Managing Configuration Files” chapter of the *Configuration Fundamentals Configuration Guide* for details on setting the CONFIG_FILE variable.



Note

When you use the **boot system** and **boot config** global configuration commands, you affect only the running configuration. You must save the environment variable settings to your startup configuration to place the information under ROM monitor control and for the environment variables to function as expected. Enter the **copy system:running-config nvram:startup-config** command to save the environment variables from your running configuration to your startup configuration.

You can view the contents of the BOOT and CONFIG_FILE environment variables using the **show bootvar** command. This command displays the settings for these variables as they exist in the startup configuration as well as in the running configuration if a running configuration setting differs from a startup configuration setting.

This example shows how to check the BOOT and CONFIG_FILE environment variables:

```
Router# show bootvar
BOOT variable = disk0:c6sup22-jsv-mz.121-5c.EX.bin,1;
CONFIG_FILE variable does not exist
Configuration register is 0x2
Router#
```

To display the contents of the configuration file pointed to by the CONFIG_FILE environment variable, enter the **more nvram:startup-config** command.