Chapter 2
Startup and Basic Configuration

2

This chapter describes basic system startup and use. Information in this chapter will help you with these tasks:

■ Using the **setup** command facility for first-time startup and basic configuration, including a step-by-step discussion of the initialization process, and a sample script of an initial configuration session.

■ Understanding and using the system's command interpreter.

■ Entering configuration commands into a configuration file, and saving and loading the configuration.

Note: Before starting up your machine, you should familiarize yourself with the attributes that are defined with the **setup** facility. Review this chapter *before* starting up your system. If you want to develop more specific controls, review the "System Configuration" chapter. You should also be familiar with the basics of using a Cisco protocol translator before attempting any configuration or management activities. The "Protocol Translator User Commands" chapter, provides information about accessing different operational modes and making connections with Cisco protocol translators.

The Setup Command Facility

The **setup** command facility enables you to start using your Cisco protocol translator quickly and without extensive background knowledge. It does this by prompting you for the information required to perform basic configuration procedures. The **setup** command facility is available on all Cisco routers, terminal servers, and protocol translators with Release 8.2 or later software.

Note: The **setup** command facility for protocol translation is available for use with Cisco's chassis-based protocol translator products. Currently, the IGS does not support use of **setup** for protocol translation specifically. Using **setup** with the IGS configures routing, as well as general system-wide characteristics. Many of these same configuration steps apply to a protocol translator. Review the basic protocol translator configuration steps outlined here to determine what configuration statements need to be included in the configuration file for an IGS that will be required to translate, as well as route and/or bridge. Refer to the "Using the DEC MOP Server" section in the "Protocol Translator User Commands" chapter, and "LAT Configuration and Management" in the "Lat Configuration and Management" chapter, for more information about configuring a protocol translator to support these features.

You use the **setup** facility both at initial system configuration and for basic changes at any time. In addition, the facility itself is a teaching tool that aids in familiarizing you with the expected command sequence as you step through the process. Because of these unique characteristics, Cisco refers to **setup** as a *command facility* rather than simply as a command.

You can use the **setup** command facility to configure system-wide (global) and interface-specific characteristics. The configuration activities associated with the **setup** process are outlined below.

These global configuration activities are available with **setup**:

■ Establishing host names

■ Setting privileged-level passwords

■ Setting virtual terminal passwords

■ Enabling SNMP network management

■ Configuring XRemote server location

These interface-specific configuration activities are handled with **setup**:

■ Configuring the Internet Protocol (IP)

■ Enabling the DEC LAT protocol

■ Enabling the DEC MOP protocol

 For more advanced applications, you must enter a privileged-level configuration session, discussed later in this chapter.

Getting Help

At any time during use of the **setup** command facility, you may request help by typing a question mark (?) at a given prompt. If the prompt requires a "yes" or "no" answer, no further help is available. However, if the prompt offers a range of acceptable answers, the help facility will give you some guidance. For example, if you are prompted to select the number of bits in a subnet mask, the help facility might inform you that you can choose a decimal number between 0 and 16.

Using Default Values

For many of the prompts, default answers appear in square brackets following the question. These defaults are used when you answer the question by pressing the Return key. If the system was previously configured, the defaults that appear are the currently configured values. If this is the first time the system has been configured, the factory defaults are provided. If there is no factory default, as in the case of passwords, nothing is displayed after the question. In the descriptions that follow, the factory defaults are shown.

How Does the Setup Command Work?

The **setup** command facility knows what kind of network interface is installed and prompts for configuration information as appropriate.

Note: Once you start the **setup** command sequence, the system runs through the entire configuration process; you cannot quit the process. If you want to make a change or correct a mistake, press the Return key or answer "no" through each of the prompts, then enter **setup** to restart the command.

When you first power up your console and protocol translator, a script similar to the following will appear on the screen. The first section of the script displays the banner information, including the software version.

```
PT Software (PT3-L), Version 9.0
Copyright (c) 1986-1992 by Cisco Systems, Inc.
Compiled Fri 17-Apr-92 11:21
```

The next portion of the display is a list of the installed hardware and software options. By reading the installed hardware, the system automatically presents the appropriate interfaces during the configuration process.

```
CSC3 (68020) processor with 4096K bytes of memory.
1 MCI Controller
1 Ethernet/IEEE 802.3 interface.
96 terminal lines
```

```
32K bytes of non-volatile configuration memory.
```

Note: The first two sections of the configuration script (the banner and the installed hardware) appear only at initial system startup. On subsequent uses of the **setup** command facility, interaction begins with the System Configuration Dialog message.

The System Configuration Dialog message and configuration steps automatically appear next, offering the prompts for which you will provide the answers to configure your system. Press the Return key to accept the default settings (if any are available). The System Configuration Dialog message provides the following initial information:

```
                      --- System Configuration Dialog ---

   At any point you may enter a question mark '?' for help.
   Refer to the 'Getting Started' Guide for additional help.
   Default settings are in square brackets '[]'.

   Continue with configuration dialog? [yes]:
```

Press the Return key to continue with the setup dialog; enter "no" to terminate the process at this point.

At any `--More--` prompts, press the space bar to continue the display.

Note: The *Getting Started* guide referenced in the System Configuration Dialog message addresses procedures for router/bridge products. For protocol translators, all startup activities are covered within this chapter.

The remainder of the script is the actual configuration process, with each prompt appearing automatically. The following paragraphs briefly describe each prompt provided during the setup process. A sample setup script is provided at the end of this section.

Answering the Setup Prompts

Configuration using the **setup** command facility has two main steps:

Step 1: Specify information for global, or system-wide parameters.

Step 2: Configure the network interfaces, based on the information previously provided.

The step-by-step discussions that follow briefly define the global and interface parameters that can be set using the **setup** command facility.

Note: More advanced configuration activities must be addressed with specific configuration procedures as described in the "System Configuration" chapter, and protocol-specific chapters of this manual.

Configuring Global Parameters

The following global configuration activities are handled via the **setup** command:

■   Defining host name

■   Setting the enable password

■   Setting the virtual terminal password

■   Specifying SNMP use

■   Defining XRemote font servers

Each step in the global configuration **setup** process is described sequentially in the descriptions that follow.

Enter host name: [PT]

Specify the name of the host computer for the protocol translator. This host name is used in the system prompts. You must enter a host name in order to continue.

Help: Type a string, then press Return. A host name can consist of from 1 to 80 alphanumeric characters and is case-sensitive.

Enter enable password:

Your system has two levels of security: user and privileged. The commands available at the user level are a subset of those available at the privileged level. Because many of the privileged commands are used to set operating parameters, these are password-protected to prevent unauthorized use. This is accomplished by setting up an *enable* password.

An enable password allows a user access to the privileged command level. You must enter an enable password in order to continue.

Default: Blank initially

Help: Type a string, then press Return. An entry must be made in order to continue with configuration.

An enable password can consist of from 1 to 80 alphanumeric characters and is case-sensitive; a number cannot be the first character. Spaces are also valid password characters; for example, "two words" is a valid password. For more information, refer to the "System Configuration" chapter.

Enter virtual terminal password:

In addition to the enable password, each virtual terminal line may also have an individual password. By default, virtual terminals require a password. If you do not set a password, the system will respond to attempted connections by displaying an error message and closing the connection. (Virtual terminals are used when remotely accessing the protocol translator via the network.)

The virtual terminal password is case-sensitive. If desired, you can allow line connections without a password. Refer to "Establishing Line Passwords" in the "System Configuration" chapter, for more information.

Default: Blank initially

Help: Type a string, then press Return.

A virtual terminal password can consist of from 1 to 80 alphanumeric characters and is case-sensitive; a number cannot be the first character. Spaces are also valid password characters; for example, "two words" is a valid password.

Configure SNMP Network Management? [yes]:

The Simple Network Management Protocol (SNMP) is the most widely supported open standard for network management. It provides a means to access and set configuration and runtime parameters of the routers and communications servers. SNMP defines a set of functions that may be used to monitor and control network elements. Answer "yes" to configure SNMP on a system-wide basis. Refer to "Configuring the Simple Network Management Protocol," in the "System Configuration" chapter, for details about specific SNMP parameters.

Answer "no" if you do not want to run SNMP.

Note: SNMP is configured on a global basis only; it cannot be specified per individual interface.

Default: Yes

Configure XRemote font servers? [no]:

Cisco supports the Network Computing Device, Inc. XRemote feature on all terminal servers and protocol translators. When fonts used by X applications are not included in ROM on the NCD terminal, one or more font servers must be identified from which the protocol translator and the NCD terminal can obtain non-ROM fonts. Answer "yes" to specify one or more XRemote font servers. Refer to the "XRemote Configuration and Managment" chapter, for details about specific XRemote parameters. Answer "no" if you do not want to specify XRemote font servers.

Default: No

Enter a font server IP address or press RETURN to exit:

A "yes" response to the previous prompt displays this prompt. Font servers must be reachable and IP-addressable entities. In order to obtain fonts from a specific host (or hosts), you must enter the IP addresses.

Enter the Internet address of the font server in dotted decimal form (for example, 131.108.2.5). If you do not know the Internet address, consult your network manager.

Note: Refer to the next section for IP address conventions.

This step is equivalent to specifying the IP address of a font server when using the **xremote tftp host** global configuration command. However, a host name cannot be entered when **setup** is running, since the name service facility is not yet running. Refer to the "XRemote Configuration and Management" chapter for more details about XRemote configuration options.

Respond to the prompt by pressing Return to stop entering font server IP addresses.

Default: None

Help: Enter an Internet address of the form "X.X.X.X", where each letter corresponds to a decimal number between 0 and 255.

Configuring Interface Parameters

The following interface configuration activities are handled via the **setup** command:

- Defining an IP address for the network interface port
- Establishing a subnet mask
- Activating DEC LAT protocol
- Activating DEC MOP protocol

Each interface configuration step in the **setup** process is described sequentially in the descriptions that follow.

Note: When a new interface is added, the **setup** command facility only prompts for information regarding the new interface. The rest of the configuration is left intact.

Configuring interface Ethernet0:
The **setup** command facility automatically presents the installed interface for you to configure.

Note: For protocol translators that have a serial network or Token Ring interface, rather than an Ethernet interface, the initial prompt specifies that port instead of an Ethernet port; otherwise the interface configuration steps for the **setup** command are the same as with an Ethernet interface.

IP address for this interface:

Internet addresses are 32-bit quantities assigned to hosts that use the TCP/IP protocols. These addresses are usually written as four decimal numbers separated by periods (for example, 192.17.5.100). Each number is an eight-bit value between 0 and 255.

The addresses are divided into three classes; the classes differ in the number of bits allocated to the network and host portions of the address.

- The Class A Internet address format allocates the highest eight bits to the network field and sets the highest-order bit to zero. The remaining 24 bits form the host field.

- The Class B Internet address allocates the highest 16 bits to the network field and sets the two highest-order bits to 1, 0. The remaining 16 bits form the host field.

- The Class C Internet address allocates the highest 24 bits to the network field and sets the three highest-order bits to 1,1,0. The remaining eight bits form the host field.

Enter your Internet address in dotted-decimal format (for example, *131.108.2.5*). If you do not know your Internet address, consult your network manager.

Table 1-1   Summary of IP Address Fields

Class	First Byte	Network Bits	Host Bits
A	1-126	8	24
B	128-191	16	16
C	192-223	24	8

Default: None

Help: Enter an Internet address using the form "X.X.X.X", where each letter corresponds to a decimal number between 0 and 255.

Number of bits in subnet field [0]:

Subnetting is an extension of the Internet addressing scheme, which allows multiple, physical networks to exist within a single Class A, B, or C network. The usual practice is to use a few of the leftmost bits in the host portion of the network address for a subnet field. The subnet field, or mask, controls whether or not subnetting is in effect on a network.

Internet addressing conventions allow either 24, 16, or 8 bits for a subnet mask, depending on whether your address is Class A, B, or C, respectively. However, because the last two of these bits are used for the host address, the **setup** command facility allows you to specify up to 22, 14, or 6 bits for your subnet mask.

A caveat applies to all classes of addresses: users cannot specify "1" as a number of bits for the subnet mask. Such a specification would provide either an all zeros or all ones subnet mask, both of which are reserved by Internet conventions.

Default: 0

Help: A decimal number between 0 and 22, 14, or 6, depending on whether your network is Class A, B, or C, respectively.

Configure LAT on this interface? [yes]:

LAT is by default enabled on any network interface for a protocol translator (if the LAT option has been purchased). Refer to the "LAT Configuration and Management" chapter for details about specific LAT parameters. Enter "yes" to enable LAT (or simply press Return).

Answer "no" if you do not want to enable LAT.

Default: Yes

Use this configuration? [yes/no]:

The system requires that you provide an answer. Answer "yes" and the configuration is stored in nonvolatile memory. Answer "no" and the configuration is not saved. This marks the end of the configuration process.

Default: None

Note: There is no default for this final prompt (requiring acknowledgment of acceptance or rejection of this configuration); you must answer either "yes" or "no" as to whether you will use this configuration.

Press RETURN to get started!

You are now ready to try using the other commands available to you, including the **show** and **configure** commands. To make changes to the configuration using the **setup** command facility, type **setup** at the EXEC prompt. Refer to later sections in this chapter and the protocol-specific chapters of this manual for advanced configuration.

Physical Configuration

The **setup** command facility operates automatically the first time you power up your protocol translator. To use **setup** on subsequent occasions, you must invoke it as you would any other command, by typing it at the EXEC, privileged level, prompt (described later in this chapter). Your system will also automatically run the **setup** program when it cannot find its configuration in memory.

Before you start using the **setup** command facility, you need to do the following:

Step 1:   Attach an RS-232 ASCII terminal to the system console port located at the rear of the Cisco server (front of the STS-10X).

Refer to the hardware publication for your server for details about cabling considerations and establishing electrical connections.

Step 2:   Configure the terminal to operate at: 9600 baud, 8 data bits, no parity, 2 stop bits.

Step 3:   Power up the protocol translator (the **setup** program runs automatically for initial startup).

Note: Network connections are not required in order to effectively use the **setup** command facility.

The server will prompt you for specific parameters including network numbers, IP addresses, and subnet masks, when applicable.

Sample Setup Configuration Script

The following is a sample of a **setup** session, including the prompts provided after **setup** has completed.

```
               --- System Configuration Dialog ---

At any point you may enter a question mark '?' for help.
Refer to the 'Getting Started' Guide for additional help.
Default settings are in square brackets '[]'.

Continue with configuration dialog? [yes]:
  Enter host name [PT]:gate-1
  Enter enable password:password
  Enter virtual terminal password:password
  Configure SNMP Network Management? [yes]:
  Configure Xremote font servers? [no]:

Configuring interface Ethernet0:
    IP address for this interface:131.108.161.10
    Number of bits in subnet field [0]:8
    Class B network is 131.108.0.0, 8 subnet bits; mask is 255.255.255.0
  Configure LAT on this interface? [yes]:

Configuring interface Serial0:
  IP address for this interface:131.108.161.50
  Number of bits in subnet field [0]:8
  Class B network is 131.108.0.0, 8 subnet bits; mask is 255.255.255.0
  Configure LAT on this interface? [yes]:

The following configuration command script was created:

hostname gate-1
enable password password
line vty 0 9
password password
snmp-server community
!
!
interface Ethernet0
ip address 131.108.161.28 255.255.255.0
lat enabled
mop enabled
!
interface Serial0
ip address 131.108.161.60 255.255.255.0
lat enabled
end

Use this configuration? [yes/no]:yes

Press RETURN to get started!

[OK]

Use the enabled mode 'configure' command to modify this configuration.

gate-1#
```

The EXEC Command Interpreter

Before exploring the commands provided with Cisco protocol translators, you should take time to familiarize yourself with the basic command line user interface integral to all Cisco products. This section introduces you to the basic Cisco protocol translator user interface environment. This environment is generally referred to as the EXEC command interpreter (or EXEC).

The EXEC interprets the commands you type and carries out the corresponding operations. You can type commands when you see the system prompt, which is the host name followed by an angle bracket (>).

Entering Commands

You may type commands in uppercase, lowercase, or both uppercase and lowercase letters. You may also abbreviate commands and other keywords to the number of characters that cause the command to be a unique abbreviation. For example, you can abbreviate the **show** command to **sh**.

If you make a typing mistake, you can erase characters with the Delete or the Backspace key. Press either key to erase the last character you typed. To erase the entire line, type Ctrl-U. (This notation means "Hold down the Ctrl key and press the U key.")

The protocol translator acts on most commands after you press the Return key. You can abort a command at any time by typing Ctrl-C.

Note: All standard Cisco control key conventions are described in Table 2-2.

To list available EXEC commands, type a question mark (**?**). You can often enter a question mark (help command) to obtain more information about commands. For example, type **terminal ?** to obtain a list of **terminal** commands or **show ?** to obtain a list of **show** commands.

Certain EXEC commands produce multiple screens of output. At the end of each screen, the EXEC pauses and displays:

```
    –More–
```

Type a space to continue the output; type anything else to return to the system command prompt.

Accessing Command Levels

For security purposes, the EXEC has two levels of access: *user* and *privileged.* The commands available at the user level are a subset of the commands available at the privileged level. Because many of the privileged commands set protocol translator operating parameters, the privileged level should be password-protected to prevent its unauthorized use. The system prompt for the privileged level ends with a pound sign (#) instead of an angle bracket (>).

The **enable** command allows access to the privileged level, prompting for a password if one has been set with the **enable-password** configuration command. For more information, see the "Establishing Passwords and System Security (TACACS)" section in the "System Configuration" chapter. To list available EXEC commands, use the **?** (question mark) command. At the user level, the **?** command produces this list:

```
cpt>?
connect <host>  Connect to host - same as typing just a host name
disconnect <cn> Break the connection specified by name or number
exit, quit      Exit from the EXEC
lat <service>   Connect to service using DEC LAT protocol
lock            Lock the terminal
name-connection Give a connection a logical name
pad <host>      Connect to host using X.29/X.3 protocols
resume          Make the named connection be current
rlogin <host>   Connect to host using rlogin protocol
show <cmd>      Information commands, type "show ?" for list
slip <addr>     Enter SLIP mode
systat          Show terminal lines and users
telnet <host>   Connect to host using telnet protocol
tn3270 <host>   Connect to host using telnet protocol (3270)
terminal        Change terminal's parameters, type "terminal ?"
where           Show open connections
x3              Set X.3 parameters on PAD
xremote         Enter XRemote mode
<cr>            To resume connection
```

At the privileged level, the **?** command lists the full protocol translator command set:

```
cpt#?
clear           Reinitialization functions, type "clear ?" for list
configure       Configure from terminal or over network
connect <host>  Connect to host - same as typing just a host name
copy            copy flashmem->tftp or tftp->flashmem
debug           Enable debugging functions, type "debug ?" for list
disable         Turn off privileged commands
disconnect <cn> Break the connection specified by name or number
enable          Turn on privileged commands
exit, quit      Exit from the EXEC
lat <service>   Connect to service using DEC LAT protocol
lock            Lock the terminal
name-connection Give a connection a logical name
pad <host>      Connect to host using X.29/X.3 protocols
ping            Send echo messages
reload          Halt and reload system
resume          Make the named connection be current
rlogin <host>   Connect to host using rlogin protocol
send <line>|*   Send message to a terminal line or lines
setup           Initialize system configuration
show <cmd>      Information commands, type "show ?" for list
```

```
slip <addr>      Enter SLIP mode
systat           Show terminal lines and users
telnet <host>    Connect to host using telnet protocol
tn3270 <host>    Connect to host using telnet protocol (3270)
terminal         Change terminal's parameters, type "terminal ?"
test             Run hardware tests, type "test ?"
trace <address>  Trace route to <address>
undebug          Disable debugging functions, type "undebug ?" for list
where            Show open connections
write            Write configuration memory, type "write ?" for list
x3               Set X.3 parameters on PAD
xremote          Enter XRemote mode
<cr>             To resume connection
```

The Configure Command

Use the privileged EXEC command **configure** to begin configuration of the protocol translator.

Begin by entering the privileged level of the EXEC. This is done by entering the **enable** command at the EXEC prompt:

    PT>**enable**

The EXEC then prompts you for the privileged level password:

    Password:

Type in the password. For security purposes, the password will not be displayed. (Also note that the password is case-sensitive.) When you enter the correct password, the system displays the privileged-level prompt:

    PT#

To begin configuration mode, enter the **configure** command at the privileged-level prompt:

    PT#**configure**

When you enter this command, the EXEC prompts you for the source of the configuration subcommands.

    Configuring from terminal, memory, or network [terminal]?

The default is to type in commands from the terminal console. Pressing the Return key begins this configuration method. Each configuration technique (terminal, memory, and network) is described in more detail later in this chapter.

The EXEC provides you with a simple editor for entering the configuration commands, and explains the editing functions:

    Enter configuration commands, one per line.
    Edit with DELETE, CTRL/W, and CTRL/U;end with CTRL/Z

Table 1-2 lists the edit key functions and their meanings.

Table 1-2   Configuration Edit Keys

Key	Meaning
Delete or Backspace	Erases one character.
Ctrl-W	Erases a word.
Ctrl-U	Erases a line.
Ctrl-R	Redisplays a line.
Return	Executes single-line commands.
Ctrl-Z	Ends configuration mode and returns to the EXEC.

Note: Use the **disable** EXEC command to return to the user command mode.

Configuration Command Types

The configuration commands are categorized by these functions:

■   Global configuration commands—Define system-wide parameters.

■   Interface subcommands—Define the characteristics of an interface (a serial or Ethernet interface, for example) and must be preceded by an **interface** command.

■   Line subcommands—Define the characteristics of a serial terminal line and must be preceded by a **line** command.

The descriptions of the commands include the command type and give examples of their use.

As with EXEC commands, you can type configuration subcommands in uppercase letters, lowercase letters, or both. You may also shorten all commands and other keywords to unique abbreviations. You may add comments by preceding the line with an exclamation point (!). Comments do not affect command processing.

If you make a typing mistake, use the Delete or Backspace key to erase a character, Ctrl-W to erase a word, and Ctrl-U to erase a line. To redisplay a line, use Ctrl-R. See Table 1-2 for a list of valid commands.

The protocol translator executes single-line commands when you press the Return key. The protocol translator does not display confirmation messages as it executes the commands. If the protocol translator encounters a problem, it displays an error message on the console terminal. When you type Ctrl-Z, the protocol translator exits the configuration mode.

In most cases, you can negate a configuration subcommand or restore a default by typing **no** before the subcommand keyword. You can usually omit the arguments of the subcommand when you negate it with **no.** The command descriptions note any exceptions to these rules.

Following are some examples of configuration files to illustrate how to enter the configuration commands.

Global Configuration Commands

Global configuration commands enable functions that affect the entire system rather than a particular line or interface, and can appear any place within the configuration file. An example of this is the global configuration command to define the host name, or the name of the protocol translator:

```
hostname translator-1
```

Interface Subcommands

Interface subcommands modify the operation of an interface such as an Ethernet, FDDI, or serial port. Interface subcommands always follow an **interface** command that defines the interface type.

The following example illustrates how to enable LAT and DEC MOP on interface *Ethernet 0*:

```
interface ethernet 0
mop enabled
lat enabled
```

Note: The EXEC accepts commands in uppercase and lowercase letters. Exclamation points (!) are not parsed and serve as comment lines and delimiters between configuration commands.

If you forget to enter the **interface** command, the system displays the message "must specify a network interface."

Line Subcommands

Line subcommands modify the operation of a serial terminal line. Line subcommands always follow a **line** command which defines the line number. If you forget to enter the **line** command, the system displays the message "must specify a line or range of lines."

The following example illustrates how to set the password on line 5:

```
line 5
password secretword
```

Type Ctrl-Z to end your configuration sessions, and the **disable** command to leave privileged level mode.

The Configuration File

If you use the **setup** facility's interactive dialog prompts to start your configuration file, it is saved in nonvolatile memory when you finish the prompts. If you chose not to create your configuration file this way, you may now choose from several options to create the configuration file.

The protocol translator holds configuration information in two places—in *running memory*, and in *nonvolatile memory*. Configuration information in running memory is temporary and will not be stored if power is shut off. Configuration information in nonvolatile memory is always available.

You use the EXEC command **write memory** to copy current (running) configuration information to nonvolatile memory. This command stores all nondefault configuration information as configuration commands in text format. The command also records a checksum for the information to protect against data corruption.

The EXEC command **show configuration** displays information stored in nonvolatile memory. You can use this command and the **write terminal** command to find differences between the current configuration (that in running memory) and that stored in nonvolatile memory. You use the EXEC command **write erase** to clear the contents of nonvolatile memory.

The **write** commands create their output by examining the state of the system currently running. The output produced by the **write** commands is generated by the software, and will not necessarily match the text the user entered to create the current configuration.

The protocol translator also allows you to store the configuration file on a network host. (This allows you to use an editor on the host to edit and create the configuration file.) Use the EXEC command **write network** to copy the current configuration information to a server host on the network. This command is described later in this section.

Configuring from the Console

To issue configuration commands from the console terminal, enter the EXEC command **configure** at the privileged-level EXEC prompt and enter configuration mode.

The protocol translator responds with a prompt asking you to specify the terminal, a file, or nonvolatile memory as the source of configuration commands.

```
Configuring from terminal, memory, or network [terminal]?
```

To begin configuration, type **terminal** at the prompt or just press Return (since terminal is the default) to start command collection.

During command collection, the protocol translator accepts one configuration command per line. You can enter as many configuration subcommands as you want.

Type Ctrl-Z when you finish entering configuration commands. This returns you to the EXEC where you can test your configuration, or write the configuration commands to memory.

At periodic intervals, you need to write the configuration information into nonvolatile memory or to a configuration file stored on a remote host. This will make checking, adding information to, and booting the configuration file an easier task. The procedures for writing information to nonvolatile memory are described next.

Writing the Configuration File to Nonvolatile Memory

After you enter the desired configuration information at the console terminal, use the privileged EXEC command **write memory** to make a copy of the configuration information in nonvolatile memory. Nonvolatile memory stores the current configuration information in text format as configuration commands, recording only nondefault settings. A checksum of memory is done to guard against corrupted data.

As part of its startup sequence, the protocol translator startup software always checks for configuration information in the nonvolatile memory. Once nonvolatile memory holds valid configuration commands, the protocol translator executes the commands automatically at startup. If the protocol translator detects a problem with the nonvolatile memory or the configuration information it contains, the protocol translator may enter the setup mode, prompting for configuration information. Problems can include a bad checksum for the information in the nonvolatile memory and the absence of critical information.

To display the configuration information stored in nonvolatile memory, enter the **show configuration** EXEC command at the privileged-level EXEC prompt.

To clear the contents of nonvolatile memory, enter the **write erase** EXEC command at the privileged-level EXEC prompt.

To re-execute the configuration commands stored in nonvolatile memory, enter **memory** at the configure mode prompt:

```
Configuring from terminal, memory, or network [terminal]?memory
```

Writing the Configuration File to a Remote Host

To store configuration information on a remote host, enter the privileged EXEC command **write network**. This command sends a copy of the current configuration information to a remote host. The command will prompt you for the destination host's address and a file name, as the following example illustrates.

Example:
```
Tokyo#write network
Remote host [131.108.2.155]?
Name of configuration file to write [tokyo-confg]?
Write file tokyo-confg on host 131.108.2.155? [confirm]y
Writing tokyo-confg...
 [OK]
```

To retrieve and/or add to the configuration information stored on a host file on a device on your network, enter **network** at the configure-mode prompt (see "The Configure Command" section in this chapter for more information):

```
Configuring from terminal, memory, or network [terminal]?network
```

The system will ask you to select a host or network configuration file. It will also ask you for the address of the host and for a file name. The following example illustrates this process.

Example:

```
Host or network configuration file [host]?
IP address of remote host [255.255.255.255]? 131.108.2.155
Name of configuration file [tokyo-confg]?
Configure using tokyo-confg from 131.108.2.155? [confirm]y
Booting tokyo-confg from 131.108.2.155: !! [OK - 874/16000 bytes]
```

Setting Up Auto Loading of the Configuration File

The protocol translator may be configured to automatically load additional configuration information from a network host. You may want to keep an up-to-date version of configuration information on another host, where you can change it as necessary, and use the nonvolatile memory as a bootstrap or backup mechanism. You can instruct the protocol translator software to load configuration information over the network by entering the **service config** subcommand and then writing the information to nonvolatile memory using the **write memory** command. Loading configuration information over the network is the default if nonvolatile memory is not installed. (The **service** configuration subcommand is described in "Tailoring Use of Network Services" in the "System Configuration" chapter.)

After loading configuration information from the nonvolatile memory, the protocol translator will attempt to load two configuration files from remote hosts. The first is the network configuration file, which contains commands that apply to all protocol translators and terminal servers on a network. The second is the host configuration file, which contains commands that apply to one protocol translator in particular.

The default name of the network configuration file is *network-confg*. The default name for the host configuration file is taken from the host name. The host name can be specified by the **hostname** configuration subcommand or can be derived from the Domain Name System (DNS); see the section "Setting the Host Name" in the "System Configuration" chapter for more information. To form the host configuration file name, the protocol translator converts the host name stripped of any DNS information to lower case, and appends "-confg." If no host name information is available, the default host configuration file name is *cpt-confg*. Other names for these configuration files can be set using the **boot** command, which is described in "Setting Configuration File Specifications" in the "System Configuration" chapter.

The protocol translator uses TFTP to load and save configuration files. By default, the protocol translator uses an Internet address of all ones to broadcast TFTP Read Request messages. However, many hosts use an old style of broadcast address consisting of all zeros. You can change operation to accommodate hosts using the old style of broadcast address.

Note: TFTP is the IP Trivial File Transfer Protocol, defined in RFC 783. The details of setting up a TFTP server process and installing the configuration files on the server host vary from one operating system to another; see the documentation for your host computer if you need more information about TFTP support.

If the protocol translator fails to load a configuration file during startup, it tries again every ten minutes (default setting) until a host provides the requested files. With each failed attempt, the protocol translator displays a message on the console terminal.

Example:

If the protocol translator is unable to load the file named `network-confg`, it displays the following message.

```
Booting network-confg... [timed out]
```

To end these file load attempts, enter the following configuration command at the console terminal and save it in the nonvolatile memory:

```
no service config
```

This command prevents the protocol translator from trying to access nonexistent TFTP servers when it is booted.

Note: Be aware that the system treats network and host configuration files differently when loading new parameters. When a host configuration file is loaded, all terminal line parameters are cleared before setting any new parameters. When a network configuration file is loaded, no old parameters are cleared. This means that terminal line parameters set by the network configuration file, which are generally loaded first, will be reset by the host configuration file, which is generally loaded second.

Software Netbooting (Downloading) Considerations

As configured at the factory, the operating system software executes instructions in the onboard EPROM. If you have a CSC/3 CPU card, you need not change the system EPROMs with each software update. Instead, you can download the latest software over the network. This process is called *netbooting*.

Netbooting works as follows: when you power up your Cisco protocol translator product for the first time, it checks the processor configuration register or the nonvolatile memory for special netbooting instructions. If the system finds no special instructions, it executes the default EPROM software.

If the system finds netbooting instructions, it determines its interface address and then runs a special process to load the new software into memory.

You can specify boot loading in two ways—by resetting bits in the processor configuration register and by configuring specific instructions using the **boot** command.

The first way involves setting the low four bits of the processor configuration register (refer to the hardware publication for your particular server product for details). If no bits are set, you must manually boot the system using the System Bootstrap program.

If only the low bit is set, the system runs the default software. The system interprets any other binary bit combination as an octal number for use in forming the boot file name. The system forms the boot file name by starting with the word *cisco* and then appending the octal number, a hyphen, and the processor type name. The System Bootstrap program displays the processor type name at system startup.

For example, if bit one in the four-bit field is set and the processor type is CSC/3, the boot file name formed is *cisco3-csc3*. Assuming no other information is available, the system would try to TFTP-load the file *cisco3-csc3* by first sending a broadcast TFTP read request to determine which server host had the file.

The second way to specify boot loading uses the nonvolatile memory option, which enables you to provide more detailed instructions for software downloading. You can use the **boot** configuration command to specify both the boot file name and the IP address of the server host. You must, however, still set the bottom four bits of the configuration register to a net-booting value.

Booting Considerations

By default, the protocol translator software continues sending TFTP Read Request messages until it receives a response. The protocol translator remains unusable as long as the network or the host with the specified file is unavailable. To limit the number of netbooting attempts, set bit 13 in the processor configuration register to 1. The software then gives up after five netbooting attempts and returns to the ROM operating software.

The protocol translator software can use any network interface, regardless of the media type or encapsulation method, to load operating software. If the interface does not support broadcasts (for example, a protocol translator using the X.25 interface does not), you must use the nonvolatile memory to specify the address of the server host with the desired file.

To display the Internet address of the server host that provided the current operating software, use the EXEC command **show version**.

Monitoring the TFTP Transaction

The TFTP session can sometimes fail. To help determine why a TFTP session failed, TFTP generates an "E" character if it receives an erroneous packet, and an "O" character if it receives an out-of-sequence packet. A period (.) indicates a timeout. The transfer session may still succeed even if TFTP generates these characters, but the output is useful for diagnosing the transfer failure.

Reloading the Operating System

To reload the operating system, use the **reload** EXEC command. The command syntax is:

reload

The **reload** command halts the system. If the system is set to restart on error, it reboots itself.

Using the Flash Memory Card for Storing and Booting System Software

With a CSC-MC+ Flash Memory Card and a CSC-MCI controller and appropriate cables, system software images may be written to Flash memory for booting. The Flash Memory Card is available for the A, C, and M chassis platforms. Refer to the section "Storing the System Image in Flash Memory," in the "System Configuration" chapter, for a description of the Flash configuration commands, and to the *Hardware Installation and Reference* publication for your particular product for hardware installation instructions. The following sections describe the commands to copy images to Flash memory.

Copying the TFTP Image to Flash Memories

The **copy tftp flash** command copies a TFTP image into the current Flash configuration:

copy tftp flash

Note: The TFTP image copied to the Flash memories must be at least system software version 9.0 or above. If system software below this version is copied into the Flash memories, the CSC-MC+ card will not be recognized by the processor card upon the next reboot.

If the image fits into the current Flash configuration, it prompts you for the IP address of the TFTP server and the TFTP file name. You are given an option to erase existing Flash memory before writing onto it. The system then clears and initializes each Flash memory. A pound sign (#) prompt is displayed for each cleared and initialized device (16 total). The entire copying process takes several minutes and will differ from network to network.

The following shows sample output of copying a system image, which is named *pt3-rx2*, into the current Flash configuration.

Example:

```
PT#copy tftp flash
IP address or name of remote host [255.255.255.255]? server1
Name of file to copy ? pt3-rx2
Copy pt3-rx2 from 150.136.128.191 into flash memory?  [confirm]
Erasure is needed before flash may be written.
Erase flash before writing? [confirm]
Clearing and initializing flash memory (please wait) ###############

Loading from 150.136.128.191: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! [OK – 1648952/4194176
bytes]
Verifying via checksum...
vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
vvvvvvvvvvvvvvvvvvvvvvv

Flash verification successful.  Length = 1648952, checksum = 0x00C2
```

The series of pound signs (#) indicate each Flash device is being cleared and initialized; one per device (16 total). The last line in the sample configuration indicates that the copy is successful. During the actual copy process, the yellow LED on the CSC-MC+ will be lighted.

The exclamation points indicate the copy process.

The series of Vs in the above sample output indicate that a checksum verification of the Flash is occurring as it is loaded into memory for boot. It is verified only through data compare during programming of the Flash.

Note: Should you wish to abort this copy process, press the Ctrl and Shift and 6 keys simultaneously. The process will abort, however the data that was copied before the abort was issued, will remain until the entire Flash memory is erased. Individual files *cannot* be erased from Flash memories.

If the TFTP image is too large, a failure message of "buffer overflow - xxxx/xxxx" will appear, where xxxx/xxxx is the number of bytes read in/number of bytes available.

Use the output of **show flash all** to obtain the image name. See "Displaying Flash ROM Statistics" in the "System Management" chapter, for more information about this EXEC command.

Copying the Flash Memory Image to a TFTP Server

You can copy an image back to a TFTP server using the **copy flash tftp** command. This copy of the system image can serve as a back-up copy and may also be used to verify that the copy in Flash is the same as on the original file on disk. The following example illustrates how to use this command:

```
PT#copy flash tftp
IP address of remote host [255.255.255.255]? 101.2.13.110
filename to write on tftp host? ptxx2
writing ptxx2 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
successful tftp write.
PT#
```

Once you have configured Flash memory, you may want to configure the system with the **no boot system flash** command in order to revert to booting from ROM. See "Storing the System Image in Flash Memory" in the "System Configuration" chapter.