

Chapter 3

Using Terminals

3

This chapter describes use of physical and virtual terminals on the Cisco router/bridge product. These tasks include:

- Making Telnet server connections from a console attached to the router/bridge
- Using the DEC MOP terminal server
- Making local changes to the terminal parameters

This chapter concludes with an alphabetical summary of the commands described in this chapter.

Making and Managing Terminal Connections

A TCP/IP Telnet connection is the basic way to communicate from a terminal to a host on a network. The Cisco Systems network servers provide Telnet communication as defined in RFC 854 and the MIL STD 1782 specification.

Making Telnet Connections

To start a Telnet connection, type a host name or a dotted-decimal Internet address at the EXEC prompt. You may precede the host name or Internet address with the command **connect** or **telnet**. This can be helpful if the host name you want to use conflicts with a network server command name.

The network server automatically numbers connections for you. Several commands use these numbers to identify connections, and you can display them using the **where** command described later in this chapter.

If you use a host name, the network server must first find the corresponding Internet address. To find this address, the network server searches its host-name-to-address cache. If the name is not in the cache, the network server uses a dynamic name lookup method. This method enables the network server to query a set of server hosts for the address.

As an option, you can specify a decimal TCP port number after the host name or Internet address when starting a Telnet connection. Normally, the network server uses the default Telnet server port, port number 23 (decimal).

After the network server determines the Internet address, or if you specify the address directly, the network server attempts to connect with the Telnet server port at that address. If the connection attempt fails, the network server displays a message to that effect and returns to the EXEC interpreter.

If the connection attempt succeeds, you can communicate with the server host as a terminal of that host. When you log off the host, the network server returns to the EXEC interpreter.

Example:

To connect to a host named *router-1*, type that name at the prompt, as seen in this example.

```
Router>router-1
```

This example illustrates how to connect to a router with IP address *103.81.25.2*:

```
Router>connect 103.81.25.2
```

Establishing Multiple Connections

The network server provides an escape sequence with which you can leave a Telnet connection without terminating it and return to the EXEC interpreter. This allows you to have any number of concurrent Telnet connections open, and to switch back and forth between them. Follow these steps to switch between connections:

Step 1: Type the escape sequence, which is usually the default key sequence Ctrl-^ X. This sequence is entered by pressing the Ctrl and ^ keys simultaneously, letting go, then pressing the X key.

Step 2: At the system command prompt, type the command to open another connection. To make a new connection, use the procedure described in the previous section, “Making Telnet Connections.” To return to an existing connection, use the **resume** command. Use the **where** command to show your open connections.

You can change the first part of the escape sequence with the **escape-character** command; see the section “Setting the Escape Character” in the chapter “Configuring the System.”

Listing Connections

Use the following command to get a listing of connections:

where

This command displays information about open connections associated with the current terminal line, as seen in the following example:

```
Router>where
```

Conn	Host	Address	Byte	Idle	Conn Name
1	DREGGS	130.106.19.50	0	0	DREGGS
2	EMBER	130.106.20.33	0	0	EMBER
* 3	CLASH	130.106.21.24	0	0	CLASH

The information includes the connection number, host name, address, number of characters waiting to be sent to the terminal, idle time, and connection name. An asterisk (*) indicates the current connection.

Resuming a Previous Connection

Use the EXEC **resume** command to resume a connection. This command has the following syntax:

```
resume [connection]
```

This command provides three ways to resume a previous connection:

- Typing the **resume** command with a connection number
- Typing only the connection number
- Pressing the Return key to return to the most recent connection.

The **where** command provides the connection number.

The following examples demonstrate use of the **resume** command.

Examples:

This command resumes connection 2.

```
Router>resume 2
```

You can omit the command name and simply type the connection number to resume that connection. This example resumes connection 3.

```
Router>3
```

To resume the most recent connection, press the Return key.

Naming a Connection

To name a connection, use the following command:

```
name-connection
```

This command assigns a logical name to a connection. The EXEC prompts for the connection number and name to assign when you enter this command. The **where** command displays a list of the assigned logical connection names.

Exiting a Session

To exit a session, use one of the following commands:

exit
quit

The **exit** and **quit** commands terminate the incoming connection and all outgoing connections from the network server. Enter one of these commands when you are finished with all sessions.

Disconnecting

To disconnect from a specified connection, use the following command:

disconnect [*connection*]

The optional argument *connection* is a connection name or number; the default is the current connection.

Do not use the **disconnect** command to end a session. Instead, log off the host, thus allowing the host to initiate the disconnect. If you cannot log off the host, then use the **disconnect** command.

Resetting a Line

To reset a terminal line, use the following privileged EXEC command:

clear line *line-number*

This command aborts any connections, terminates the associated processes, and resets the data structures associated with a terminal line.

The argument *line-number* specifies the terminal line number.

Executing Special Telnet Commands

The Cisco Telnet software supports special Telnet commands in the form of Telnet sequences that map generic terminal control functions to operating system-specific functions.

To issue a special Telnet command, type the escape sequence (usually Ctrl-^) and then a command character. You can type the command character as you hold down Ctrl or with Ctrl released; you can type either uppercase or lowercase letters.

Table 1-1 lists the special Telnet commands.

Table 1-1 Special Commands in the Cisco Systems Telnet Implementation

Command Name	Command Characters
Break	Ctrl-^, B
Interrupt Process (IP)	Ctrl-^, C
Erase Character (EC)	Ctrl-^, H
Abort Output (AO)	Ctrl-^, O
Are You There? (AYT)	Ctrl-^, T
Erase Line (EL)	Ctrl-^, U

At any time during an active Telnet session, you can list the Telnet commands by typing this command at the system prompt:

Ctrl ^ ?

This is done by typing the escape sequence followed by a question mark. This command displays an online table of the special Telnet commands, for quick reference.

A sample of this table follows (the Ctrl key is represented by the first ^ character).

```
[Special telnet escape help]
^^B  sends telnet BREAK
^^C  sends telnet IP
^^H  sends telnet EC
^^O  sends telnet AO
^^T  sends telnet AYT
^^U  sends telnet EL
```

Incoming Telnet Connections

In addition to the console terminal, each network server supports up to five incoming Telnet connections. Each of these connections can start an EXEC interpreter process on the network server.

The user of an incoming Telnet connection can gain access to the privileged EXEC commands through the **enable** command, which requires a password. With access to the complete EXEC command set, the incoming connection acts as a remote console. A remote console connection provides a convenient way to monitor and adjust network server operation.

You can control access to the network server with access lists; see the section “Configuring IP Access Lists” in the chapter “Routing IP” for more information.

The network server supports the following Telnet options:

- Echo

- Binary Transmission
- Suppress Go Ahead
- Terminal Type
- Send Location

Displaying TCP Connections

To show the status of a TCP connection, enter this EXEC command:

```
show tcp [line-number]
```

The **show tcp** command displays the status of all TCP connections. Specify the optional argument *line-number* in octal to display the status of the TCP connections for a particular line. The following example shows the command output:

```
con0 (console terminal), connection 1 to host MATHOM
Connection state is ESTAB, I/O status: 1, unread input bytes: 1
Local host: 192.31.7.18, 33537 Foreign host: 192.31.7.17, 23
Enqueued packets for retransmit: 0, input: 0, saved: 0
Event Timers (current time is 2043535532):
Timer:          Retrans  TimeWait  AckHold    SendWnd    KeepAlive
Starts:          69      0         69         0          0
Wakeups:         5       0         1          0          0
Next:          2043536089      0         0          0          0
iss: 2043207208 snduna: 2043211083  sndnxt: 2043211483   sndwnd: 1344
irs: 3447586816 rcvnxt: 3447586900  rcvwnd:      2144 delrcvwnd: 83
RTTO: 565 ms, RTV: 233 ms, KRTT: 0 ms, minRTT: 68 ms, maxRTT: 1900 ms
ACK hold: 282 ms
Datagrams (max data segment is 536 bytes):
Rcvd: 106 (out of order: 0), with data: 71, total data bytes: 83
Sent: 96 (retransmit: 5), with data: 92, total data bytes: 4678
```

Displaying Active Sessions

To show the current active sessions, use the following command:

```
show sessions
```

The **show sessions** command provides information about open Telnet connections. This command may be run at the user-level prompt.

Displaying Information About Active Lines

The **show users** and **systat** EXEC commands display information about the active lines of the network server, including the line number, connection names, and terminal location.

```
show users [all]
```

```
systat [all]
```

Specify the optional keyword **all** to display information for both active and inactive lines. These commands enable monitoring of virtual terminal use. You may issue these commands at the user-level prompt. They are synonymous.

Changing Terminal Parameters

The following sections describe how to change the terminal parameters using the **terminal** commands. The new settings temporarily override those made with the line configuration subcommands described in the section “Configuring Console and Virtual Terminal Lines” in the chapter “Configuring the System.”

To obtain information about the terminal configuration parameter settings for the current terminal line, use the **show terminal** command. To display information about the active ports of the server, use the **show users** command. The information displayed includes the line number, connection name, idle time, and terminal location.

Some **terminal** commands use the decimal representation of an ASCII character as an argument. See the appendix “ASCII Character Set” for ASCII-to-decimal conversion information.

To display a list of commands that you can enter to change the hardware and software parameters of the current terminal line, use the command:

terminal ?

Each command has a **no** variation that undoes the local setting.

Changing the Terminal Screen Width

To set or unset the number of characters (columns) on a single line of the current terminal screen, use the **terminal width** command:

terminal width *columns*
terminal no width

The login protocol uses the argument *columns* to set up terminal parameters on a remote host.

Example:

This example sets the terminal width to 132 columns.

```
Router>terminal width 132
```

Changing the Terminal Screen Length

To set or unset the number of lines on the screen of the current terminal, use the **terminal length** command:

```
terminal length screen-length  
terminal no length
```

The argument *screen-length* is the desired number of lines.

The server uses this value to determine when to pause during multiple-screen output. The default length is 24 lines. A value of zero disables pausing between screens of output. The screen length specified can be learned by hosts.

Changing the Terminal Escape Character

To set or remove the escape character for the current terminal line, use the **terminal escape-character** command:

```
terminal escape-character decimal-number  
terminal no escape-character
```

The argument *decimal-number* is the ASCII decimal representation of the desired escape character or an escape sequence (Ctrl-P, for example). Typing the escape character followed by the X key returns you to the EXEC when you are connected to another computer. The default escape character is Ctrl-^ X. (See the appendix “ASCII Character Set, for a list of ASCII characters.)

The operating software interprets the Break key on the console as an attempt to halt the system.

Note: Depending upon the configuration register setting, console breaks will either be ignored or cause the server to shut down. The Break key cannot be used as the escape character on the Cisco router.

Example:

This example sets Ctrl-P as the escape character:

```
Router>terminal escape-character 17
```

Displaying the Debug Messages on the Console and Terminals

To display the debug message on the console and terminals, use the **terminal monitor** command:

terminal monitor
terminal no monitor

The **terminal monitor** command copies **debug** command output and system error messages to the current terminal as well as to the console terminal.

To use this command, you must first issue the **enable** command and enter the password to access the privileged command mode.

Establishing Input Notification

To establish input notification, use the **terminal notify** command:

terminal notify
terminal no notify

When you have multiple concurrent connections, you may want to know when output is pending on a connection other than the current connection. For example, you may want to know when another connection receives mail or a message. The **terminal notify** command causes the router to notify you of pending output. The **terminal no notify** command ends such notifications.

Changing the Character Padding

To set the character padding on the current terminal line, use the **terminal padding** commands:

terminal padding *decimal-number count*
terminal no padding *decimal-number*

The argument *decimal-number* is the ASCII decimal representation of the character. The argument *count* is the number of NULL bytes sent after that character. (The appendix “ASCII Character Set” provides a list of the ASCII characters.)

The **terminal no padding** command ends this padding.

Example:

This example pads RETURNS (ASCII character 25) with 20 NULL bytes:

```
Router>terminal padding 25 20
```

Displaying Terminal Parameter Settings

To display the configuration parameter settings for the current terminal, use this EXEC command:

show terminal

This command may be issued at the user-level prompt.

Using the DEC MOP Server

All Cisco internetworking products include a server that implements a subset of Digital Equipment Corporation's (DEC's) Maintenance Operation Protocol (MOP) for Ethernet interfaces. The MOP server supports the request ID message, periodic system ID messages, and the remote console carrier functions.

The MOP server periodically multicasts a system ID message, which is used by DEC's Ethernet configurator to determine what stations are present in an Ethernet network. The configurator is controlled by the Network Control Program (NCP) command **define module configurator**. For more information on this command, consult DECnet-VAX documentation.

The Cisco internetworking products use the MOP communication device code of 121. This code has been assigned to Cisco by DEC, although some versions of DECnet-VAX software may report the code numerically, rather than with a device name. The Digital Ethernet configurator product also makes use of receipt of system ID messages when building network maps.

The MOP server supports the DEC remote console function. In this capacity, a system manager on a DECnet system that does not include TCP/IP can create a virtual terminal connection to a Cisco router. The NCP commands **connect node** and **connect via** are used to connect to the remote console. Due to the nature of the MOP server, only a single inbound connection per Ethernet interface is supported. The MOP server does not contain the necessary mechanisms for supporting more than one connection at a time.

MOP is not a routable protocol. To bridge the MOP console carrier and system ID functions, enable bridging for protocol type 6002. The periodic system ID messages are sent to the multicast address *AB00.0002.0000*.

The EXEC command **debug mop** reports events occurring within the MOP server, including reception of request ID messages, transmission of system ID messages, and reservation and release of the remote console.

Enabling MOP for an Interface

To control whether MOP is enabled for an interface, use the **mop enabled** interface sub-command.

mop enabled
no mop enabled

The default is enabled. Use the **no mop enabled** command to disable MOP.

Controlling the MOP System ID Messages

To control whether MOP periodic system ID messages are sent out to an interface, use the **mop sysid** interface subcommand.

mop sysid
no mop sysid

You can still run MOP without having the background system ID messages sent out. This lets you use the MOP remote console, but does not generate messages used by the configurator.

Use the **no mop sysid** to disable MOP from sending the system ID messages.

EXEC Terminal Command Summary

This section lists the EXEC commands described in this chapter in alphabetical order.

clear line *line-number*

Aborts connections and processes and resets the line. Argument *line-number* specifies the line.

{**connect** | **telnet**} [*connection*]

Either of the commands connects to a remote host using the Telnet protocol. The optional argument *connection* specifies a host name or an IP address.

Ctrl ^ ?

Lists the special Telnet commands in the form of Telnet sequences that map generic terminal control functions to operating system-specific functions. Type this command at the system prompt.

disconnect [*connection*]

Closes the specified connection. The optional argument *connection* is a connection name or number; the default is the current connection.

exit
quit

Either of these commands terminates the EXEC command processor and closes any active Telnet sessions.

name-connection

Assigns a logical name to a connection. The EXEC prompts for the connection number and name to assign when you enter this command.

resume [*connection*]

Resumes a connection. The optional argument *connection* is a connection name or number.

show sessions

Provides information about open Telnet connections.

show tcp [*line-number*]

Displays the status of all TCP connections. Specify the optional argument *line-number* in octal to display the status of the TCP connections for a particular line.

show terminal

Displays information about the terminal configuration parameter settings for the current terminal line and the active ports of the server. The optional keyword **all** requests information for both active and inactive ports.

show users [**all**]**systat** [**all**]

Displays information about active lines. The optional **all** keyword provides information about inactive as well as active ports.

terminal ?

Lists commands you can enter to change hardware and software parameters of the current line.

terminal [**no**] **escape-character** *decimal-number*

Sets the escape character for the current terminal line. The argument *decimal-number* is either the ASCII decimal representation of the desired escape character or an escape sequence. The default escape character is Ctrl-^ X. The **no** version of the command removes the specified escape character for the current terminal line.

terminal [no] length *screen-length*

Sets the length of the terminal to *screen-length* lines. The default length is 24 lines. The **no** version of the command unsets the number of lines.

terminal [no] monitor

Displays the debug message on the console and terminals. It copies **debug** command output and system error messages to the current terminal as well as to the console terminal. To use this command, you must first issue the **enable** command and enter the password to access the privileged command mode. The **no** version of the command discontinues the display of the debug message.

terminal [no] notify

Enables notification to a terminal when output to it is generated on any of its active sessions. The **no** version of this command disables notification.

terminal [no] padding *decimal-number count*

Sets the character padding on the current terminal line. The argument *decimal-number* is the ASCII decimal representation of the character. The argument *count* is the number of NULL bytes sent after that character. (The appendix “ASCII Character Set” provides a list of the ASCII characters.) The **terminal no padding** command ends this padding.

terminal [no] width *columns*

Sets the number of characters (columns) on a single line of the current terminal screen. The login protocol uses the argument *columns* to set up terminal parameters on a remote host. The **no** version of the command unsets the number of characters (columns) on the single line of the current terminal screen.

where

Displays information about open connections associated with the current terminal line and provides the connection number.

