



# Configuring Operating Characteristics for Terminals

---

This chapter describes how to configure operating characteristics for terminals. For a complete description of the terminal operation commands in this chapter, refer to the “Terminal Operating Characteristics Commands” chapter in the *Cisco IOS Release 12.1 Configuration Fundamentals Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

To configure operating characteristics for terminals, perform any of the tasks in the following sections:

- Displaying Information about Current Terminal Session
- Setting Local Terminal Parameters
- Saving Local Settings Between Sessions
- Ending a Session
- Changing Terminal Session Parameters
- Recording the Device Location
- Changing the Retry Interval for a Terminal Port Queue
- Configuring LPD Protocol Support

## Displaying Information about Current Terminal Session

To display terminal line information, use either of the following commands in user or privileged EXEC mode:

Command	Purpose
Router> <code>show whoami text</code>	Displays information about the current user's terminal line, including hostname, line number, line speed, and location. If text is included as an argument in the command, that text is displayed as part of the additional data about the line.
Router> <code>where</code>	Lists all open sessions associated with the current terminal line. An asterisk (*) in the output indicates the current terminal session.

The following example shows sample output of the **show whoami** command:

```
Router> show whoami

Comm Server "Router", Line 0 at 0bps. Location "Second floor, West"

--More--
Router>
```

To prevent the information from being lost, this command always displays a More prompt before returning. Press the space bar to return to the prompt.

## Setting Local Terminal Parameters

The **terminal EXEC** commands enable or disable features for the current session only. You can use these commands to temporarily change terminal line settings without changing the stored configuration file.



### Note

References to EXEC mode indicate either user EXEC mode or privileged EXEC mode.

To see a list of the commands for setting terminal parameters for the current session, use the following command in User or Privileged EXEC Mode:

Command	Purpose
Router> <b>terminal ?</b>	Lists the commands for setting terminal parameters.

The following example shows sample output for the **terminal ?** command. Commands available on your router will vary depending on the software image and hardware you are using.

```
Router> terminal ?
 autohangup           Automatically hangup when last connection closes
 data-character-bits  Size of characters being handled
 databits             Set number of data bits per character
 dispatch-character   Define the dispatch character
 dispatch-timeout     Set the dispatch timer
 download             Put line into 'download' mode
 editing              Enable command line editing
 escape-character     Change the current line's escape character
 exec-character-bits  Size of characters to the command exec
 flowcontrol          Set the flow control
 full-help            Provide help to unprivileged user
 help                 Description of the interactive help system
 history              Enable and control the command history function
 hold-character       Define the hold character
 ip                   IP options
 keymap-type          Specify a keymap entry to use
 lat                  DEC Local Area Transport (LAT) protocol-specific
                    configuration
 length               Set number of lines on a screen
 no                   Negate a command or set its defaults
 notify               Inform users of output from concurrent sessions
```

padding	Set padding for a specified output character
parity	Set terminal parity
rxspeed	Set the receive speed
special-character-bits	Size of the escape (and other special) characters
speed	Set the transmit and receive speeds
start-character	Define the start character
stop-character	Define the stop character
stopbits	Set async line stop bits
telnet	Telnet protocol-specific configuration
telnet-transparent	Send a CR as a CR followed by a NULL instead of a CR followed by a LF
terminal-type	Set the terminal type
transport	Define transport protocols for line
txspeed	Set the transmit speeds
width	Set width of the display terminal

Throughout this chapter, many terminal settings can be configured for all terminal sessions or for just the current terminal session. Settings for all terminal sessions are configured in line configuration mode and can be saved. Settings for the current session are specified using EXEC mode commands which generally begin with the word **terminal**.

## Saving Local Settings Between Sessions

You can configure the Cisco IOS software to save local parameters (set with **terminal** EXEC commands) between sessions. Saving these local settings ensures that the parameters the user sets will remain in effect between terminal sessions. This function is useful for servers in private offices. To save local settings between sessions, use the following command in line configuration mode:

Command	Purpose
Router(config-line)# <b>private</b>	Saves local settings between sessions.

If the **private** line configuration command is not used, user-set terminal parameters are cleared when the session ends with either the **exit** EXEC command, or when the interval set with the **exec-timeout** line configuration command has passed.

## Ending a Session

To end a session, use the following command:

Command	Purpose
Router> <b>quit</b>	Exits the active session.

Refer to the “Managing Connections, Menus, and System Banners” chapter for more information on exiting sessions and closing connections.

# Changing Terminal Session Parameters

This section explains how to change terminal and line settings both for a particular line and locally. The local settings are set with the EXEC **terminal** commands. They temporarily override the settings made by the system administrator and remain in effect only until you exit the system. In line configuration mode, you can set terminal operation characteristics that will be in operation for that line until the next time you change the line parameters.

The following sections describe the more common changes to the terminal and line settings:

- Defining the Escape Character and Other Key Sequences
- Specifying Telnet Operation Characteristics
- Configuring Data Transparency for File Transfers
- Specifying an International Character Display

The following sections describe the less common changes to the terminal and line settings:

- Setting Character Padding
- Specifying the Terminal and Keyboard Type
- Changing the Terminal Screen Length and Width
- Changing Pending Output Notification
- Creating Character and Packet Dispatch Sequences
- Displaying Debug Messages on the Console and Terminals
- Changing Flow Control for the Current Session
- Setting a Terminal-Locking Mechanism
- Configuring Automatic Baud Rate Detection
- Setting a Line as Insecure
- Configuring Communication Parameters for Terminal Ports

Further discussion of Terminal and Line settings specific to dial-up routers (Access Servers) can be found in the *Cisco IOS Dial Services Configuration Guide: Terminal Services* document, available on Cisco.com.

## Defining the Escape Character and Other Key Sequences

You can define or modify the default keys used to execute functions for system escape, terminal activation, disconnect, and terminal pause. Generally, the keys used are actually combinations of keys, such as pressing the Control (Ctrl) key and another key (or keys) at the same time (such as **Ctrl-^**). Sequences of keys, such as pressing the Control key and another key, then pressing yet another key, are also sometimes used (for example **Ctrl-^, x**). However, in each case these keys are referred to as characters, because each key or combination of keys is represented by a single ASCII character. For a complete list of available ASCII characters and their decimal and keyboard equivalents, see the “ASCII Character Set” appendix of the *Cisco IOS Release 12.1 Configuration Fundamentals Command Reference*.

## Globally Defining Escape Character and Other Key Sequences

To define or change the default key sequence, use one or more of the following commands in line configuration mode:

Command	Purpose
Router(config-line)# <b>escape-character</b> { <i>ascii-number</i>   <i>character</i>   <b>break</b>   <b>default</b>   <b>none</b> } [ <b>soft</b> ]	Changes the system escape character. We recommend the use of the ASCII characters represented by the decimal numbers 1 through 30. The escape character can be a single character (such as !), a key combination (such as <b>Ctrl-X</b> ), or a sequence of keys (such as <b>Ctrl-^, X</b> ). The default escape character (key combination) is <b>Ctrl-Shift-6</b> (Ctrl-^), or <b>Ctrl-Shift-6, X</b> (Ctrl-^, X).
Router(config-line)# <b>activation-character</b> <i>ascii-number</i>	Defines a session activation character. Entering this character at a vacant terminal begins a terminal session. The default activation character is the Return key.
Router(config-line)# <b>disconnect-character</b> <i>ascii-number</i>	Defines the session disconnect character. Entering this character at a terminal ends the session with the router. There is no default disconnect character.
Router(config-line)# <b>hold-character</b> <i>ascii-number</i>	Defines the hold character that causes output to the screen to pause. To continue the output, enter any character after the hold character. To use the hold character in normal communications, precede it with the escape character. There is no default hold character.

For most of the above commands, you can reinstate the default value for the escape character or activation character by using the **no** form of the command. However, to return the escape-character to its default, you should use the **escape-character default** command.



### Note

If you are using the **autoselect** function, the activation character should not be changed from the default value of Return. If you change this default, the **autoselect** feature may not function immediately.

## Defining Escape and Pause Characters for the Current Session

For the current terminal session, you can modify key sequences to execute functions for system escape and terminal pause. To modify these sequences, use one or more of the following commands in EXEC mode:

Command	Purpose
Router> <b>terminal escape-character</b> <i>ascii-number</i>	Changes the system escape sequence for the current session. The escape sequence indicates that the codes that follow have special meaning. The default key combination is <b>Ctrl-Shift-6</b> (Ctrl-^).
Router> <b>terminal hold-character</b> <i>ascii-number</i>	Defines the hold sequence or character that causes output to the terminal screen to pause for this session. There is no default sequence. To continue the output, type any character after the hold character. To use the hold character in normal communications, precede it with the escape character. You cannot suspend output on the console terminal.

The **terminal escape-character** command is useful, for example, if you have the default escape character defined for a different purpose in your keyboard file. Entering the escape character followed by the X key returns you to EXEC mode when you are connected to another device.

## Specifying Telnet Operation Characteristics

The following sections discuss telnet operation characteristics tasks:

- Generating a Hardware Break Signal for a Reverse Telnet Connection
- Setting the Line to Refuse Full-Duplex, Remote Echo Connections
- Allowing Transmission Speed Negotiation
- Synchronizing the Break Signal
- Changing the End-of-Line Character

### Generating a Hardware Break Signal for a Reverse Telnet Connection

To cause the router to generate a hardware Break signal on the EIA/TIA-232 line that is associated with a reverse Telnet connection for the current line and session, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal telnet break-on-ip</b>	Generates a hardware Break signal on the EIA/TIA-232 line that is associated with a reverse Telnet connection for the current line and session.

The hardware Break signal occurs when a Telnet Interrupt-Process command is received on that connection. This command can be used to control the translation of Telnet IP commands into X.25 Break indications.

This command is also a useful workaround in the following situations:

- Several user Telnet programs send an Interrupt-Process command, but cannot send a Telnet Break signal.
- Some Telnet programs implement a Break signal that sends an Interrupt-Process command.

Some EIA/TIA-232 hardware devices use a hardware Break signal for various purposes. A hardware Break signal is generated when a Telnet Break command is received.



**Note** This command applies only to access server products. It is not supported on stand-alone routers.

## Setting the Line to Refuse Full-Duplex, Remote Echo Connections

You can set the line to allow the Cisco IOS software to refuse full-duplex, remote echo connection requests from the other end. This refusal suppresses negotiation of the Telnet Remote Echo and Suppress Go Ahead options. To set the current line to refuse to negotiate full-duplex for the current session, remote echo options on incoming connections, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal telnet refuse-negotiations</b>	Sets the current line to refuse to negotiate full-duplex for the current session.



**Note** This command applies only to access server products. It is not supported on stand-alone routers.

## Allowing Transmission Speed Negotiation

To allow the Cisco IOS software to negotiate transmission speed for the current line and session, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal telnet speed</b> <i>default-speed</i> <i>maximum-speed</i>	Allows the Cisco IOS software to negotiate transmission speed for the current line and session.

You can match line speeds on remote systems in reverse Telnet, on host machines that connect to the network through an access server, or on a group of console lines hooked up to an access server, when disparate line speeds are in use at the local and remote ends of the connection. Line speed negotiation adheres to the Remote Flow Control option, defined in RFC 1080.



**Note** This command applies only to access server products. It is not supported on stand-alone routers.

## Synchronizing the Break Signal

You can set lines on Access Servers to cause a reverse Telnet line to send a Telnet Synchronize signal when it receives a Telnet Break signal. The TCP Synchronize signal clears the data path, but interprets incoming commands. To cause the Cisco IOS software to send a Telnet Synchronize signal when it receives a Telnet Break signal on the current line and session, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal telnet sync-on-break</b>	Causes the Cisco IOS software to send a Telnet Synchronize signal when it receives a Telnet Break signal on the current line and session.



### Note

This command applies only to Access Server products. It is not supported on stand-alone routers.

## Changing the End-of-Line Character

The end of each line typed at the terminal is ended with a Return (CR). To cause the current terminal line to send a CR as a CR followed by a NULL instead of a CR followed by a line feed (LF), use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal telnet transparent</b>	Causes the current terminal line to send a CR as a CR followed by a NULL instead of a CR followed by a line feed (LF).

This command ensures interoperability with different interpretations of end-of-line handling in the Telnet protocol specification.



### Note

This command applies only to access servers. It is not supported on stand-alone routers.

## Configuring Data Transparency for File Transfers

Data transparency enables the Cisco IOS software to pass data on a terminal connection without the data being interpreted as a control character.

During terminal operations, some characters are reserved for special functions. For example, **Ctrl-Shift-6-X** (^X) suspends a session. When transferring files over a terminal connection (using the Xmodem or Kermit protocols, for example), you must suspend the recognition of these special characters to allow a successful file transfer. This process is called *data transparency*.

You can set a line to act as a transparent pipe so that programs such as Kermit, Xmodem, or CrossTalk can download a file across a terminal line. To temporarily configure a line to act as a pipe for file transfers, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal download</b>	Sets up the terminal line to act as a transparent pipe for file transfers.

The **terminal download** command is equivalent to using all the following commands.

- **terminal telnet transparent**
- **terminal no escape-character**
- **terminal no hold-character**
- **terminal no padding 0**
- **terminal no padding 128**
- **terminal parity none**
- **terminal databits 8**

## Specifying an International Character Display

The classic U.S. ASCII character set is limited to 7 bits (128 characters), which adequately represents most displays in the U.S. Most defaults on the modem router work best on a 7-bit path. However, international character sets and special symbol display can require an 8-bit wide path and other handling.

You can use a 7-bit character set (such as ASCII), or you can enable a full 8-bit international character set (such as ISO 8859). This allows special graphical and international characters for use in banners and prompts, and adds special characters such as software flow control. Character settings can be configured globally, per line, or locally at the user level. Use the following criteria for determining which configuration mode to use when you set this international character display:

- If a large number of connected terminals support nondefault ASCII bit settings, use the global configuration commands.
- If only a few of the connected terminals support nondefault ASCII bit settings, use line configuration commands or the EXEC local terminal setting commands.



### Note

Setting the EXEC character width to an 8-bit character set can cause failures. If a user on a terminal that is sending parity enters the **help** command, an “unrecognized command” message appears because the system is reading all eight bits, although the eighth bit is not needed for **help**.



### Note

If you are using the **autoselect** function, the activation character should be set to the default Return, and the EXEC character bit should be set to 7. If you change these defaults, the application does not recognize the activation request.

## Specifying the International Character Display for All Lines

To specify a character set for all lines, use one or both of the following commands in global configuration mode:

Command	Purpose
Router(config)# <b>default-value exec-character-bits</b> {7   8}	Specifies the character set used in EXEC and configuration command characters.
Router(config)# <b>default-value special-character-bits</b> {7   8}	Specifies the character set used in special characters such as software flow control, hold, escape, and disconnect characters.

## Specifying the International Character Display on a Hardware, Software, or Per-line Basis

To specify a character set based on hardware, software, or on a per-line basis, use any of the following commands in line configuration mode:

Command	Purpose
Router(config-line)# <b>databits</b> {5   6   7   8}	Sets the number of data bits per character that are generated and interpreted by hardware.
Router(config-line)# <b>data-character-bits</b> {7   8}	Sets the number of data bits per character that are generated and interpreted by software.
Router(config-line)# <b>exec-character-bits</b> {7   8}	Specifies the character set used in EXEC and configuration command characters on a per-line basis.
Router(config-line)# <b>special-character-bits</b> {7   8}	Specifies the character set used in special characters such as software flow control, hold, escape, and disconnect characters on per-line basis.

## Specifying an International Character Display for the Current Session

To specify a character set based on hardware, software, or on a per-line basis for the current terminal session, use the following appropriate commands in EXEC mode:

	Command	Purpose
<b>Step 1</b>	Router> <b>terminal databits</b> {5   6   7   8}	Sets the number of data bits per character that are generated and interpreted by hardware for the current session.
<b>Step 2</b>	Router> <b>terminal data-character-bits</b> {7   8}	Sets the number of data bits per character that are generated and interpreted by software for the current session.
<b>Step 3</b>	Router> <b>terminal exec-character-bits</b> {7   8}	Specifies the character set used in EXEC and configuration command characters on a per-line basis for the current session.
<b>Step 4</b>	Router> <b>terminal special-character-bits</b> {7   8}	Specifies the character set used in special characters (such as software flow control, hold, escape, and disconnect characters) on per-line basis for the current session.

## Setting Character Padding

Character padding adds a number of null bytes to the end of the string and can be used to make a string an expected length for conformity. You can change the character padding on a specific output character.

### Globally Setting Character Padding

To set character padding, use the following command in line configuration mode:

Command	Purpose
Router(config-line)# <b>padding</b> <i>ascii-number count</i>	Sets padding on a specific output character for the specified line.

### Changing Character Padding for the Current Session

To change character padding on a specific output character for the current session, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal padding</b> <i>ascii-number count</i>	Sets padding on a specific output character for the specified line for this session.

## Specifying the Terminal and Keyboard Type

You can specify the type of terminal connected to a line. This feature has two benefits: it provides a record of the type of terminal attached to a line, and it can be used in Telnet terminal negotiations to inform the remote host of the terminal type for display management.

### Globally Specifying the Terminal Type

To specify the terminal type, use the following command in line configuration mode:

Command	Purpose
Router(config-line)# <b>terminal-type</b> { <i>terminal-name</i>   <i>terminal-type</i> }	Specifies the terminal type.

This feature is used by TN3270 terminal to identify the keymap and ttycap passed by the Telnet protocol to the end host.

## Changing the Terminal and Keyboard Type for the Current Session

To specify the type of terminal connected to the current line for the current session, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal terminal-type</b> <i>terminal-type</i>	Specifies the terminal type for this session.

Indicate the terminal type if it is different from the default of VT100. This default is used by TN3270 for display management and by Telnet and rlogin to inform the remote host of the terminal type.

To specify the current keyboard type for a session, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal keymap-type</b> <i>keymap-name</i>	Specifies the keyboard type for this session.

You must specify the keyboard type when you use a keyboard other than the default of VT100. The system administrator can define other keyboard types and give you their names.

## Changing the Terminal Screen Length and Width

By default, the Cisco IOS software provides a screen display of 24 lines by 80 characters. You can change these values if they do not meet the requirements of your terminal. The screen values you set are passed during rsh and rlogin sessions.

The screen values set can be learned by some host systems that use this type of information in terminal negotiation. To disable pausing between screens of output, set the screen length to a zero.

The screen length specified can be learned by remote hosts. For example, the rlogin protocol uses the screen length to set up terminal parameters on a remote UNIX host. The width specified also can be learned by remote hosts.

## Globally Changing the Terminal Screen Length and Width

To set the terminal screen length and width, use the following commands in line configuration mode:

	Command	Purpose
<b>Step 1</b>	Router(config-line)# <b>length</b> <i>screen-length</i>	Sets the screen length.
<b>Step 2</b>	Router(config-line)# <b>width</b> <i>characters</i>	Sets the screen width.

## Changing the Terminal Screen Length and Width for the Current Session

To set the number of lines or character columns on the current terminal screen for the current session, use one of the following commands in EXEC mode:

Command	Purpose
Router> <b>terminal length</b> <i>screen-length</i>	Sets the screen length for the current session.
Router> <b>terminal width</b> <i>characters</i>	Sets the screen width for the current session.

## Changing Pending Output Notification

You can set up a line to inform a user who has multiple, concurrent Telnet connections when output is pending on a connection other than the active one. For example, you might want to know when another connection receives mail or a message.

### Globally Setting Pending Output Notification

To set pending output notification, use the following command in line configuration mode:

Command	Purpose
Router(config-line)# <b>notify</b>	Sets up a line to notify a user of pending output.

### Setting Pending Output Notification for the Current Session

To set pending output notification for the current session, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal notify</b>	Sets up a line to notify a user of pending output for the current session.

## Creating Character and Packet Dispatch Sequences

The Cisco IOS software supports dispatch sequences and TCP state machines that transmit data packets only when they receive a defined character or sequence of characters. You can set up dispatch characters that allow packets to be buffered, then transmitted upon receipt of a character. You can set up a state machine that allows packets to be buffered, then transmitted upon receipt of a sequence of characters. This feature enables packet transmission when the user presses a function key, which is typically defined as a sequence of characters, such as “Esc I C.”

TCP state machines can control TCP processes with a set of predefined character sequences. The current state of the device determines what happens next, given an expected character sequence. The state-machine commands configure the server to search for and recognize a particular sequence of characters, then cycle through a set of states. The user defines these states—up to eight states can be defined. (Think of each state as a task that the server performs based on the assigned configuration commands and the type of character sequences received.)

The Cisco IOS software supports user-specified state machines for determining whether data from an asynchronous port should be sent to the network. This functionality extends the concept of the dispatch character and allows the equivalent of multicharacter dispatch strings.

Up to eight states can be set up for the state machine. Data packets are buffered until the appropriate character or sequence triggers the transmission. Delay and timer metrics allow for more efficient use of system resources. Characters defined in the TCP state machine take precedence over those defined for a dispatch character.

### Setting Character and Packet Dispatch Sequences for a Line

Use the following commands in line configuration mode, as needed, for your particular system needs:

	Command	Purpose
Step 1	Router(config-line)# <b>state-machine</b> <i>name state firstchar lastchar [nextstate   transmit]</i>	Specifies the transition criteria for the states in a TCP state machine.
Step 2	Router(config-line)# <b>dispatch-machine</b> <i>name</i>	Specifies the state machine for TCP packet dispatch.
Step 3	Router(config-line)# <b>dispatch-character</b> <i>ascii-number [ASCII-number2 . . . ascii-number]</i>	Defines a character that triggers packet transmission.
Step 4	Router(config-line)# <b>dispatch-timeout</b> <i>milliseconds</i>	Sets the dispatch timer.
Step 5	Router(config-line)# <b>buffer-length</b> <i>length</i>	Specifies the maximum length of the data stream to be forwarded.

### Changing the Packet Dispatch Character for the Current Session

To change the packet dispatch character for the current session, use the following command in EXEC mode:

Command	Purpose
Router> <b>terminal dispatch-character</b> <i>ascii-number1 [ascii-number2 . . . ascii-number]</i>	Defines a character that triggers packet transmission for the current session.

## Displaying Debug Messages on the Console and Terminals

To display **debug** command output and system error messages in EXEC mode on the current terminal, use the following command in privileged EXEC mode:

Command	Purpose
Router# <b>terminal monitor</b>	Displays debug command output and system error messages in EXEC mode on the current terminal.

Remember that all terminal parameter-setting commands are set locally and do not remain in effect after a session is ended. You must use this command at the privileged-level EXEC prompt at each session to see the debugging messages.

## Changing Flow Control for the Current Session

To configure flow control between the router and attached device for this session, use one of the following commands in EXEC mode:

Command	Purpose
Router> <b>terminal flowcontrol</b> { <b>none</b>   <b>software</b> [ <b>in</b>   <b>out</b> ]   <b>hardware</b> }	Sets the terminal flow control for this session.
Router> <b>terminal start-character</b> <i>ascii-number</i> <sup>1</sup>	Sets the flow control start character in the current session.
Router> <b>terminal stop-character</b> <i>ascii-number</i> <sup>1</sup>	Sets the flow control stop character in the current session.

1. This command is seldom used. Typically, you only need to use the **terminal flowcontrol** command.

## Setting a Terminal-Locking Mechanism

You can enable a terminal-locking mechanism that allows a terminal to be temporarily locked by using the following command in line configuration mode:

Command	Purpose
Router(config-line)# <b>lockable</b>	Enables a temporary terminal locking mechanism.

After you configure the line as lockable, you must still issue the **lock** EXEC command to lock the keyboard.

## Configuring Automatic Baud Rate Detection

You can configure a terminal to detect the baud rate being used over an asynchronous serial line automatically. To set up automatic baud detection, use the following command in line configuration mode:

Command	Purpose
Router(config-line)# <b>autobaud</b>	Sets the terminal to automatically detect the baud rate.



### Note

Do not use the **autobaud** command with the **autoselect** command.

To start communications using automatic baud detection, use multiple Returns at the terminal. A 600-, 1800-, or 19200- baud line requires three Returns to detect the baud rate. A line at any other baud rate requires only two Returns. If you use extra Returns after the baud rate is detected, the EXEC facility simply displays another system prompt.

## Setting a Line as Insecure

You can set up a terminal line to appear as an insecure dial-up line. The information is used by the LAT software, which reports such dial-up connections to remote systems.

To set a line as insecure, use the following command in line configuration mode:

Command	Purpose
Router(config-line)# <b>insecure</b>	Sets the line as a dial-up line.

In early releases of Cisco IOS software, any line that used modem control was reported as dial-up connection through the LAT protocol; this command allows more direct control of your line.

## Configuring Communication Parameters for Terminal Ports

You can change these parameters as necessary to meet the requirements of the terminal or host to which you are attached. To do so, use one or more of the following commands in EXEC mode:

Command	Purpose
Router> <b>terminal speed</b> <i>bps</i> Router> <b>terminal txspeed</b> <i>bps</i> Router> <b>terminal rxspeed</b> <i>bps</i>	Sets the line speed for the current session. Choose from line speed, transmit speed, or receive speed.
Router> <b>terminal databits</b> {5   6   7   8}	Sets the data bits for the current session.
Router> <b>terminal stopbits</b> {1   1.5   2}	Sets the stop bits for the current session.
Router> <b>terminal parity</b> {none   even   odd   space   mark}	Sets the parity bit for the current session.

## Recording the Device Location

You can record the location of a serial device. The text provided for the location appears in the output of the EXEC monitoring commands. To record the device location, use the following command in line configuration mode:

Command	Purpose
Router(config-line)# <b>location</b> <i>text</i>	Records the location of a serial device.

## Changing the Retry Interval for a Terminal Port Queue

If you attempt to connect to a remote device (such as a printer) that is busy, the connection attempt is placed in a terminal port queue. If the retry interval is set too high, and several routers or other devices are connected to the remote device, your connection attempt can have long delays. To change the retry interval for a terminal port queue, use the following command in global configuration mode:

Command	Purpose
Router(config)# <b>terminal-queue</b> <b>entry-retry-interval</b> <i>interval</i>	Changes the retry interval for a terminal port queue.

## Configuring LPD Protocol Support

The Cisco IOS software supports a subset of the Berkeley UNIX Line Printer Daemon (LPD) protocol used to send print jobs between UNIX systems. This subset of the LPD protocol permits the following:

- Improved status information
- Cancellation of print jobs
- Confirmation of successful printing and automatic retry for common print failures
- Use of standard UNIX software

The Cisco implementation of LPD permits you to configure a printer to allow several types of data to be sent as print jobs (for example, PostScript or raw text).

To configure a printer for the LPD protocol, use the following command in global configuration mode:

Command	Purpose
Router(config)# <b>printer</b> <i>printername</i> { <b>line</b> <i>number</i>   <b>rotary</b> <i>number</i> } [ <b>newline-convert</b> ]	Configures printer and specify a TTY line (or lines) for the device.

If you use the **printer** command, you also must modify the */etc/printcap* file on the UNIX system to include the definition of the remote printer on the router. Use the optional **newline-convert** keyword on UNIX systems that do not handle single character line terminators to convert a new line to a character Return, line-feed sequence.

The following example includes the configuration of the printer Saturn on the host Memphis:

```
comm1pt|Printer on cisco AccessServer:\
:rm=memphis:rp+saturan:\
:sd+/usr/spool/lpd/comm1pt:\
:lf=?var/log/lpd/comm1pt:
```

The content of the actual file may differ, depending on the configuration of your UNIX system.

To print, users use the standard UNIX lpr command.

Support for the LPD protocol allows you to display a list of currently defined printers and current usage statistics for each printer. To do so, use the following command in EXEC mode:

Command	Purpose
Router> <b>show printer</b>	Lists currently defined printers and their usage statistics.

To provide access to LPD features, your system administrator must configure a printer and assign a TTY line (or lines) to the printer. The administrator must also modify */etc/printcap* on your UNIX system to include the definition of the remote printer in the Cisco IOS software.