CHAPTER 4

Terminal Line and Modem Support Commands

The line configuration commands described in this chapter are used to configure lines on the communication server.

For line configuration information and examples, refer to the chapter "Configuring Terminal Lines and Modem Support" in the *Access and Communication Servers Configuration Guide*.

Note Many parameters that can be configured on a line can also be configured for the current session on a line as well. These temporary terminal and line-setting parameters are described in the *Cisco Access Connection Guide*.

activation-character

To define the character you type at a vacant terminal to begin a terminal session, use the **activation-character** line configuration command. Use the **no** form of this command to make any character activate a terminal.

activation-character *ascii-number*
no activation-character

Syntax Description

ascii-number	Decimal representation of the activation character

Default

Return (decimal 13)

Command Mode

Line configuration

Usage Guidelines

See the “ASCII Character Set” appendix for a list of ASCII characters.

Example

The following example sets the activation character for the console to Delete, which is Decimal 127:

```
line console
activation-character 127
```

autobaud

To set the line for automatic baud detection, use the **autobaud** line configuration command. Use the **no autobaud** command to restore the default.

autobaud
no autobaud

Syntax Description

This command has no arguments or keywords.

Default

No autobaud detection

Command Mode

Line configuration

Usage Guidelines

The autobaud detection supports a range from 300 to 19200 baud. A line set for autobaud cannot be used for outgoing connections. Nor can you set autobaud capability on a line using 19200 baud when the parity bit is set because of hardware limitations.

Example

The following example sets the auxiliary port for autobaud detection:

```
line 5
autobaud
```

autocommand

To configure the communication server to automatically execute a command or list of commands when a user connects to a particular line, use the **autocommand** line configuration command.

autocommand *command*

Syntax Description

command	Any appropriate EXEC command, including the host name and any switches that occur with the EXEC command

Default

None

Command Mode

Line configuration

Example

The following example forces an automatic connection to a host named *host21* (which could be an IP address). In addition, the UNIX UUCP application specifies TCP socket 25, and the **/stream** switch enables a raw TCP stream with no Telnet control sequences.

```
line vty 4
autocommand connect host21 uucp /stream
```

autohangup

To configure automatic line disconnect, use the **autohangup** line configuration command. This command causes the EXEC to issue the **exit** command when the last connection closes.

autohangup

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Line configuration

Usage Guidelines

This command is useful for UNIX UUCP applications that automatically disconnect lines because UUCP scripts cannot issue the **exit** command to hang up the telephone.

Example

The following example enables automatic line disconnect on lines 5 through 10:

```
line 5 10
autohangup
```

autoselect

To configure a line to start an ARA, PPP, or SLIP session, use the **autoselect** line configuration command. Use the **no** form of this command to disable this function on a line.

autoselect {**arap** | **ppp** | **slip**}
no autoselect

Syntax Description

arap	Configures the communication server to allow an ARA session to start up automatically.
ppp	Configures the communication server to allow a PPP session to start up automatically.
slip	Configures the communication server to allow a SLIP session to start up automatically.

Default

Configures the communication server to allow an ARA session to start up automatically.

Command Mode

Line configuration

Usage Guidelines

This command eliminates the need for users to enter an EXEC command to start an ARA, PPP, or SLIP session.

Note SLIP does not support authentication. For PPP, you must enable authentication. For ARA, authentication is automatically enabled.

The **autoselect** command configures the communication server to identify the type of connection being requested. For example, when a user on a Macintosh running ARA selects the Connect button, the communication server automatically starts an ARAP session. If, on the other hand, the user is running SLIP or PPP and uses the **autoselect ppp** or **autoselect slip** command, the communication server automatically starts a PPP or SLIP session, respectively. This command is used on lines used for making different types of connections.

A line that does not have **autoselect** configured will see an attempt to open a connection as noise. The communication server will not respond; the user client will time out.

Note After the modem connection is established, a Return is required to evoke a response, such as to get the username prompt. You might need to update your scripts to include this requirement.

Examples

The following example enables ARA on a line:

```
line 3
arap enable
autoselect arap
```

The following example enables PPP on a line:

```
line 7
autoselect ppp
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

ppp authentication chap†
ppp authentication pap†
arap use-tacacs†
ppp use-tacacs†

banner exec

To display a message on terminals with an interactive EXEC, use the **banner exec** global configuration command. This command specifies a message to be displayed when an EXEC process is created (a line is activated, or an incoming connection is made to a VTY).

banner exec *d message d*

Syntax Description

d	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
message	Message text.

Default

None

Command Mode

Global configuration

Usage Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

Example

The following example sets an EXEC message. The dollar sign ($) is used as a delimiting character.

```
banner exec $
Session activated. Enter commands at the prompt.
$
```

Related Commands

banner incoming
banner motd
exec-banner

banner incoming

To specify a message used when you have an incoming connection to a line from a host on the network, use the **banner incoming** global configuration command.

banner incoming *d message d*

Syntax Description

d	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
message	Message text.

Default

None

Command Mode

Global configuration

Usage Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

An incoming connection is one initiated from the network side of the communication server. The EXEC banner can be suppressed on certain lines using the **no exec-banner** line configuration command. This line should *not* display the EXEC or MOTD banners when an EXEC is created.

Example

The following example sets an incoming connection message. The pound sign (#) is used as a delimiting character.

```
banner incoming #
Welcome to Rhesus.
#
```

Related Commands

banner exec
banner motd
exec-banner

banner motd

To specify a message-of-the-day (MOTD) banner, use the **banner motd** global configuration command.

banner motd *d message d*

Syntax Description

d	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
message	Message text.

Default

None

Command Mode

Global configuration

Usage Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

This MOTD banner is displayed to all terminals connected, and is useful for sending messages that affect all users; impending system shutdowns, for example.

The **banner** command without any keywords specified defaults to the **banner motd** command. When a new **banner motd** command is added to the configuration, it overwrites the existing **banner** command (no keyword specified). Similarly, if a **banner** command is added to the configuration, any exiting **banner motd** command is overwritten.

Example

The following example sets a MOTD banner. The pound sign (#) is used as a delimiting character.

```
banner motd #
Building power will be off from 7:00 AM until 9:00 AM this coming Tuesday.
#
```

Related Commands

banner exec
banner incoming
exec-banner

busy-message

To create a "host failed" message that displays when a connection fails, use the **busy-message** global configuration command. Use the **no** form of this command to disable the "host failed" message from displaying on the specified host.

busy-message *hostname d message d*
no busy-message *hostname*

Syntax Description

hostname	Name of the host that cannot be reached.
d	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the message.
message	Message text.

Default

None

Command Mode

Global configuration

Usage Guidelines

This command applies only to Telnet connections.

Follow the **busy-message** command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

Defining a "host failed" message for a host prevents all communication server-initiated user messages, including the initial message that indicates the connection is "Trying..." The **busy-message** command can be used in the **autocommand** command to suppress these messages.

Example

The following example sets a message that will be displayed on the terminal whenever an attempt to connect to the host named *dross* fails. The pound sign (#) is used as a delimiting character.

```
busy-message dross #
Cannot connect to host. Contact the computer center.
#
```

clear line

Use the **clear line** EXEC command to return a terminal line to idle state.

clear line *line-number*

Syntax Description

line-number	Absolute line number

Default

None

Command Mode

EXEC

Usage Guidelines

Use this command to log out of a specific session running on another line. If the line uses a modem, the modem will be disconnected.

Example

In the following example, line 3 is reset to idle state:

```
clear line 3
```

databits

To set the number of data bits per character that are interpreted and generated by hardware, use the **databits** line configuration command.

databits {**5** | **6** | **7** | **8**}

Syntax Description

5	Five data bits per character
6	Six data bits per character
7	Seven data bits per character
8	Eight data bits per character

Default

8 data bits per character

Command Mode

Line configuration

Usage Guidelines

The **databits** line configuration command can be used to mask the high bit on input from devices that generate 7 data bits with parity. If parity is being generated, specify 7 data bits per character. If no parity generation is in effect, specify 8 data bits per character. The other keywords are supplied for compatibility with older devices and generally are not used.

Example

The following example changes the data bits to 7 on line 4:

```
line 4
databits 7
```

Related Commands

Two daggers (††) indicates that the command is documented in the *Cisco Access Connection Guide*.

terminal data-character-bits ††
terminal databits ††

data-character-bits

To set the number of data bits per character that are interpreted and generated by software, use the **data-character-bits** line configuration command.

data-character-bits {**7** | **8**}

Syntax Description

7	Seven data bits per character
8	Eight data bits per character

Default

8 data bits per character

Command Mode

Line configuration

Usage Guidelines

The **data-character-bits** line configuration command is used primarily to strip parity from X.25 connections on communication servers with the protocol translation software option. The **data-character-bits** line configuration command does not work on hardwired lines.

Example

The following example sets the number of data bits per character for virtual terminal line 1 to 7:

```
line vty 1
data-character-bits 7
```

default-value exec-character-bits

To define the EXEC character width for either 7 bits or 8 bits, use the **default-value exec-character-bits** global configuration command.

default-value exec-character-bits {**7** | **8**}

Syntax Description

7	Selects the 7-bit ASCII character set.
8	Selects the full 8-bit ASCII character set.

Default

7-bit ASCII character set

Command Mode

Global configuration

Usage Guidelines

Configuring the EXEC character width to 8 bits allows you to add graphical and international characters in banners, prompts, and so forth. However, setting the EXEC character width to 8 bits can also cause failures. If a user on a terminal that is sending parity enters the command **help**, an "unrecognized command" message appears because the system is reading all 8 bits, although the eighth bit is not needed for the **help** command.

Example

The following example selects the full 8-bit ASCII character set for EXEC banners and prompts:

```
default-value exec-character-bits 8
```

Related Commands

Two daggers (††) indicates that the command is documented in the *Cisco Access Connection Guide*.

default-value special-character-bits
exec-character-bits
special-character-bits
terminal exec-character-bits ††
terminal special-character-bits ††

default-value special-character-bits

To configure the flow control default value from a 7-bit width to an 8-bit width, use the **default-value special-character-bits** global configuration command.

default-value special-character-bits {**7** | **8**}

Syntax Description

7	Selects the 7-bit character set.
8	Selects the full 8-bit character set.

Default

7-bit character set

Command Mode

Global configuration

Usage Guidelines

Configuring the special character width to 8 bits allows you to add graphical and international characters in banners, prompts, and so forth.

Example

The following example selects the full 8-bit special character set:

```
default-value special-character-bits 8
```

Related Commands

Two daggers (††) indicates that the command is documented in the *Cisco Access Connection Guide*.

default-value exec-character-bits
exec-character-bits
special-character-bits
terminal exec-character-bits ††
terminal special-character-bits ††

disconnect-character

To define a character to disconnect a session, use the **disconnect-character** line configuration command. Use the **no** form of this command to remove the disconnect character.

disconnect-character *ascii-number*
no disconnect-character

Syntax Description

ascii-number	Decimal representation of the session disconnect character

Default

No disconnect character is defined.

Command Mode

Line configuration

Usage Guidelines

The Break character is represented by zero; NULL cannot be represented.

To use the session disconnect character in normal communications, precede it with the escape character. See the "ASCII Character Set" appendix for a list of ASCII characters.

Example

The following example sets the disconnect character for virtual terminal line 4 to Escape, which is decimal character 27:

```
line vty 4
disconnect-character 27
```

dispatch-character

To define a character that causes a packet to be sent, use the **dispatch-character** line configuration command. Use the **no** form of this command to remove the definition of the specified dispatch character.

dispatch-character *ascii-number1* [*ascii-number2 . . . ascii-number*]
no dispatch-character *ascii-number1* [*ascii-number2 . . . ascii-number*]

Syntax Description

ascii-number	Decimal representation of the character, such as Return (decimal 13) for line-at-a-time transmissions

Default

No dispatch character is defined.

Command Mode

Line configuration

Usage Guidelines

The **dispatch-character** command defines a dispatch character that causes a packet to be sent even if the dispatch timer has not expired. It causes the communication server to attempt to buffer characters into larger-sized packets for transmission to the remote host. The communication server normally dispatches each character as it is typed.

This command can take multiple arguments, so you can define any number of characters as dispatch characters.

Example

The following example specifies the Return character (decimal 13) as the dispatch character:

```
line vty 4
dispatch-character 13
```

Related Commands

dispatch-machine
dispatch-timeout
state-machine

dispatch-machine

To specify an identifier for a TCP packet dispatch state machine, use the **dispatch-machine** line configuration command.

dispatch-machine *name*

Syntax Description

name	Name of the state machine that determines when to send packets on the asynchronous line

Default

No dispatch state machine identifier is defined.

Command Mode

Line configuration

Usage Guidelines

When the **dispatch-timeout** command is specified, a packet being built will be sent when the timer expires, and the state will be reset to zero.

Any dispatch characters specified using the **dispatch-character** command are ignored when a state machine is also specified.

If a packet becomes full, it will be sent regardless of the current state, but the state is not reset. The packet size depends upon the traffic level on the asynchronous line, as well as the dispatch-timeout value. There is always room for 60 data bytes. If the dispatch-timeout value is greater than or equal to 100 ms, a packet size of 536 (data bytes) is allocated.

Example

The following example specifies the name *packet* for the state machine:

```
line 1 20
dispatch-machine packet
```

Related Commands

dispatch-character
dispatch-timeout
state-machine

dispatch-timeout

To set the character dispatch timer, use the **dispatch-timeout** line configuration command. Use the **no** form of this command to remove the timeout definition.

dispatch-timeout *milliseconds*
no dispatch-timeout

Syntax Description

milliseconds	Integer that specifies the number of milliseconds the communication server waits after putting the first character into a packet buffer before sending the packet. During this interval, more characters might be added to the packet, which increases the processing efficiency of the remote host.

Default

No dispatch timeout is defined.

Command Mode

Line configuration

Usage Guidelines

The **dispatch-timeout** line configuration command causes the communication server to buffer characters into packets for transmission to the remote host. The communication server sends a packet a specified amount of time after the first character is put in the buffer. The communication server normally dispatches each character as it is entered. You can use the **dispatch-timeout** and **dispatch-character** line configuration commands together. In this case, the communication server dispatches a packet each time the dispatch character is entered, or after the specified dispatch timeout interval, depending on which condition is met first.

Note The communication server's response might appear intermittent if the timeout interval is greater than 100 milliseconds and remote echoing is used.

Example

The following example sets the dispatch timer to 80 milliseconds:

```
line vty 0 4
dispatch-timeout 80
```

Related Commands

dispatch-character
dispatch-machine
state-machine

editing

To enable enhanced editing mode for a particular line, use the **editing** line configuration command. Use the **no** form of this command to disable the enhanced editing mode.

editing
no editing

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Line configuration

Usage Guidelines

Table 4-1 provides a description of the keys used to enter and edit commands. Ctrl indicates the Control key. It must be pressed simultaneously with its associated letter key. Esc indicates the Escape key. It must be pressed first, followed by its associated letter key. Keys are *not* case sensitive.

Table 4-1 Editing Keys and Functions for Software Release 9.21 and Later

Keys	Function
Tab	Completes a partial command name entry. When you enter a unique set of characters and press the Tab key, the system completes the command name. If you enter a set of characters that could indicate more than one command, the system beeps to indicate an error. Enter a question mark (?) immediately following the partial command (no space). The system provides a list of commands that begin with that string.
Delete or Backspace	Erases the character to the left of the cursor.
Return	At the command line, pressing the Return key performs the function of processing a command. At the "`---More---`" prompt on a terminal screen, pressing the Return key scrolls down a line.
Space Bar	Allows you to see more output on the terminal screen. Press the space bar when you see the line "`---More---`" on the screen to display the next screen.
Left arrow[1]	Moves the cursor one character to the left. When you enter a command that extends beyond a single line, you can press the Left Arrow key repeatedly to scroll back toward the system prompt and verify the beginning of the command entry.
Right arrow[1]	Moves the cursor one character to the right.
Up arrow[1] or Ctrl-P	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down arrow[1] or Ctrl-N	Return to more recent commands in the history buffer after recalling commands with the Up Arrow or Ctrl-P. Repeat the key sequence to recall successively more recent commands.
Ctrl-A	Moves the cursor to the beginning of the line.
Ctrl-B	Moves the cursor back one character.

Keys	Function
Ctrl-D	Deletes the character at the cursor.
Ctrl-E	Moves the cursor to the end of the command line.
Ctrl-F	Moves the cursor forward one character.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-L and Ctrl-R	Redisplays the system prompt and command line.
Ctrl-T	Transposes the character to the left of the cursor with the character located at the cursor.
Ctrl-U and Ctrl-X	Deletes all characters from the cursor back to the beginning of the command line.
Ctrl-V and Esc Q	Inserts a code to indicate to the system that the keystroke immediately following should be treated as a command entry, *not* as an editing key.
Ctrl-W	Deletes the word to the left of the cursor.
Ctrl-Y	Recalls the most recent entry in the delete buffer. The delete buffer contains the last ten items you have deleted or cut. Ctrl-Y can be used in conjunction with Esc Y.
Ctrl-Z	Ends configuration mode and returns you to the EXEC prompt.
Esc B	Moves the cursor back one word.
Esc C	Capitalizes the word at the cursor.
Esc D	Deletes from the cursor to the end of the word.
Esc F	Moves the cursor forward one word.
Esc L	Changes the word at the cursor to lowercase.
Esc U	Capitalizes from the cursor to the end of the word.
Esc Y	Recalls the next buffer entry. The buffer contains the last ten items you have deleted. Press Ctrl-Y first to recall the most recent entry. Then press Esc Y up to nine times to recall the remaining entries in the buffer. If you bypass an entry, continue to press Esc Y to cycle back to it.

[1] The arrow keys function only with ANSI-compatible terminals.

Table 4-2 lists the editing keys and functions of software releases before Software Release 9.21.

Table 4-2 Editing Keys and Functions for Software Release 9.1 and Earlier

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

Example

In the following example, enhanced editing mode is disabled on virtual terminal line 3:

```
line vty 3
no editing
```

Related Command

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal editing ††

escape-character

To define a system escape character, use the **escape-character** line configuration command. Use the **no** form of this command to set the escape character to Break.

escape-character *ascii-number*
no escape-character

Syntax Description

ascii-number	Either the decimal representation of the character or a control sequence (Ctrl-E, for example).

Default

Ctrl-^

Command Mode

Line configuration

Usage Guidelines

The Break key cannot be used as an escape character on the console terminal because the operating software interprets Break as an instruction to halt the system. To send the escape character to the other side, press **Ctrl-^** twice.

See the "ASCII Character Set" appendix for a list of ASCII characters.

Example

The following example sets the escape character to Ctrl-P, which is decimal character 16:

```
line console
escape-character 16
```

exec

To allow an EXEC process on a line, use the **exec** line configuration command. Use the **no** form of this command to turn off the EXEC process for the specified line.

exec
no exec

Syntax Description

This command has no arguments or keywords.

Default

The communication server starts EXEC processes on all lines.

Command Mode

Line configuration

Usage Guidelines

When you want to allow an outgoing connection *only* for a line, use the **no exec** command. When a user tries to Telnet to a line with the **no exec** command configured, the user will get no response when pressing the Return key at the login screen.

Example

The following example illustrates how to turn off the EXEC on line 7. You might want to do this on the auxiliary port if the attached device (for example, the control port of a rack of modems) sends unsolicited data to the communication server. An EXEC process would start if this happened, making the line unavailable.

```
line 7
no exec
```

exec-banner

To control whether banners are displayed or suppressed, use the **exec-banner** line configuration command. This command determines whether or not the communication server will display the EXEC banner or the message-of-the-day (MOTD) banner when an EXEC is created. Use the **no** form of this command to suppress the banner messages.

exec-banner
no exec-banner

Syntax Description

This command has no arguments or keywords.

Default

The messages defined with **banner motd** and **banner exec** commands are displayed on all lines.

Command Mode

Line configuration

Example

The following example suppresses the banner on virtual terminal lines 0 to 4:

```
line 0 4
no exec-banner
```

Related Commands

banner exec
banner motd

exec-character-bits

To configure the character widths of EXEC and configuration command characters, use the **exec-character-bits** line configuration command.

exec-character-bits {**7** | **8**}

Syntax Description

7	Selects the 7-bit character set.
8	Selects the full 8-bit character set for use of international and graphical characters in banner messages, prompts, and so forth.

Default

7-bit ASCII character set

Command Mode

Line configuration

Usage Guidelines

Setting the EXEC character width to 8 allows you to use special graphical and international characters in banners, prompts, and so forth. However, setting the EXEC character width to 8 bits can cause failures. If a user on a terminal that is sending parity enters the command **help**, an "unrecognized command" message appears because the system is reading all 8 bits, although the eighth bit is not needed for the **help** command.

Example

The following example allows full 8-bit international character sets by default, except for the console, which is an ASCII terminal. It illustrates use of the **default-value exec-character-bits** global configuration command and the **exec-character-bits** line configuration command.

```
default-value exec-character-bits 8
line 0
exec-character-bits 7
```

Related Commands

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

default-value exec-character-bits
default-value special-character-bits
special-character-bits
terminal exec-character-bits ††
terminal special-character-bits ††

exec-timeout

To set the interval that the EXEC command interpreter waits until user input is detected, use the **exec-timeout** line configuration command. Use the **no** form of this command to remove the timeout definition.

exec-timeout *minutes* [*seconds*]
no exec-timeout

Syntax Description

minutes	Integer that specifies the number of minutes.
seconds	(Optional) Additional time intervals in seconds. An interval of zero specifies no timeouts.

Default

10 minutes

Command Mode

Line configuration

Usage Guidelines

If no input is detected, the EXEC resumes the current connection, or if no connections exist, it returns the terminal to the idle state and disconnects the incoming session. It is the same as entering **exec-timeout 0**.

Examples

The following example sets a time interval of 2 minutes, 30 seconds:

```
line console
exec-timeout 2 30
```

The following example sets a time interval of 10 seconds:

```
line console
exec-timeout 0 10
```

flowcontrol

To set the method of data flow control between the terminal or other serial device and the communication server, use the **flowcontrol** line configuration command. Use the **no** form of this command to disable flow control.

flowcontrol {**none** | **software** [**in** | **out**] | **hardware** [**in** | **out**]}
no flowcontrol {**none** | **software** [**in** | **out**] | **hardware** [**in** | **out**]}

Syntax Description

none Turns off flow control.

software Sets software flow control. An optional keyword specifies the direction: **in** causes the communication server to listen to flow control from the attached device, and **out** causes the communication server to send flow control information to the attached device. If you do not specify a direction, both are assumed.

hardware Sets hardware flow control. An optional keyword specifies the direction: **in** causes the communication server to listen to flow control from the attached device, and **out** causes the communication server to send flow control information to the attached device. If you do not specify a direction, both are assumed. For more information about hardware flow control, see the hardware installation and maintenance manual for your communication server.

Default

No flow control

Command Mode

Line configuration

Usage Guidelines

When software flow control is set, the default stop and start characters are Ctrl-S and Ctrl-Q (XOFF and XON). You can change them with the **stop-character** and **start-character** commands.

Example

The following example sets hardware flow control on line 7:

```
line 7
flowcontrol hardware
```

Related Commands

start-character
stop-character

hold-character

To define the local hold character used to pause output to the terminal screen, use the **hold-character** line configuration command. Use the **no** form of this command to restore the default.

hold-character *ascii-number*
no hold-character

Syntax Description

ascii-number	Either the decimal representation of the hold character or a control sequence (for example, Ctrl-P)

Default

No hold character is defined.

Command Mode

Line configuration

Usage Guidelines

The Break character is represented by zero; NULL cannot be represented. 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. See the “ASCII Character Set” appendix for a list of ASCII characters.

Example

The following example sets the hold character to Ctrl-S, which is decimal 19:

```
line 8
hold-character 19
```

Related Command

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal hold-character ††

insecure

To set the line as an insecure location, use the **insecure** line configuration command. Use the **no** form of this command to disable this feature.

insecure
no insecure

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Line configuration

Example

The following example sets up line 10 as a dial-up line that is used by the LAT software to report the line as available to remote hosts:

```
line 10
insecure
```

length

To set the terminal screen length, use the **length** line configuration command.

length *screen-length*

Syntax Description

screen-length	Number of lines on the screen. A value of zero disables pausing between screens of output.

Default

24 lines

Command Mode

Line configuration

Usage Guidelines

The communication server software uses the value of this command to determine when to pause during multiple-screen output. Not all commands recognize the configured screen length. For example, the **show terminal** command assumes a screen length of 24 lines or more.

Example

The following example illustrates how to disable the screen pause function on the terminal connected to line 6:

```
line 6
terminal-type VT220
length 0
```

line

To identify a specific line for configuration and start the line configuration command collection mode, use the **line** global configuration command.

line [**aux** | **console | tty** | **vty**] *line-number* [*ending-line-number*]

Syntax Description

aux	(Optional) Auxiliary EIA/TIA-232 DTE port. Must be addressed as relative line 0. The auxiliary port can be used for modem support and asynchronous connections.
console	(Optional) Console terminal line. The console port is DCE.
tty	(Optional) Standard asynchronous line.
vty	(Optional) Virtual terminal for remote console access.
line-number	The relative number of the terminal line (or the first line in a contiguous group) you want to configure when the line type is specified. Numbering begins with zero.
ending-line-number	(Optional) The relative number of the last line in a contiguous group you want to configure. If you omit the keyword, then *line-number* and *ending-line-number* are absolute rather than relative line numbers.

Default

There is no default line.

Command Mode

Global configuration

Usage Guidelines

Virtual terminal lines are used to allow remote access to the communication server. A virtual terminal line is not associated with either the console or auxiliary port. You can address a single line or a consecutive range of lines with the **line** command. A line number is necessary, though, and you will receive an error message if you forget to include it.

Entering the **line** command with the optional line type (**aux**, **con**, **tty**, or **vty**) designates the line number as a relative line number. For example, to configure line parameters for line 7 (a tty line), you could enter the following:

```
cs(config)# line tty 7
```

You also can enter the **line** command without specifying a line type. In this case, the line number is treated as an absolute line number. For example, to configure line parameters for line 5 (could be of any type), you could enter the following:

```
cs (config)# line 5
```

Absolute line numbers increment consecutively and can be difficult to manage on large systems. Relative line numbers are a shorthand notation used in configuration. Internally, the communication server uses absolute line numbers. You cannot use relative line numbers everywhere, but you can use absolute line numbers everywhere.

The absolute line number of the auxiliary port is 1. The relative line number of the auxiliary port is 0. See the **modem** line configuration command to set up modem support on the auxiliary port. See the **modem** line configuration commands in the "Configuring Terminal Lines and Modem Support" chapter of the *Access and Communication Servers Configuration Guide*.

The software keeps a table of absolute and relative line numbers that you can display with the EXEC command **show users all**. A sample display follows:

```
cs> show users all
  Line      User     Host(s)               Idle   Location
   0 con 0                                        chaff console
   1 tty 1                                        Engineering printer
   2 tty 2
   3 tty 3           DREGGS                1:07   Katy x1111
   4 tty 4                                        Console E3-D
   5 tty 5                                        Mkt. demo area
   6 tty 6
   7 tty 7           DREGGS                  14   Marie x1112
  10 tty 10
 .  .  .
 135 tty 135
 136 tty 136
 137 tty 137                                      rp4-printer
 140 tty 140                                      Braille printer
 141 aux 0
 142 vty 0   Denise    idle                       DENISE-MAC.CISCO.COM
 143 vty 1   Michael   idle                     0 DREGGS.CISCO.COM
 144 vty 2
 145 vty 3
 146 vty 4
 147 vty 5
```

The absolute line numbers are listed at the far left, followed by the line type and then the relative line number. Relative line numbers always begin numbering at zero and define the type of line. Addressing the second virtual terminal line as line vty 1, for example, is easier than remembering it as line 143.

The line types are ranked as follows in the line table:

1. Console 0 (con 0)
2. Standard asynchronous line (tty)
3. Auxiliary port (aux)
4. Virtual terminal line (vty)
5. Printer

You can address a single line or a consecutive range of lines with the **line** command. A line number is necessary, though, and you will receive an error message if you forget to include it.

Note Line numbers are in octal form on the modular communication server products (ASM-CS), but are in decimal form on the Cisco 2500 and 500-CS communication servers.

The terminal from which you locally configure the communication server is attached to the console port. To configure line parameters for the console port, you would enter the following:

```
cs(config)# line con 0
```

The console relative line number must be 0.

Virtual terminal lines are used to allow remote access to the communication server. A virtual terminal line is not associated with either the auxiliary or console port. You can address either a single line or a consecutive range of lines. The communication server has five virtual terminal lines by default. However, you can create additional virtual terminal lines as described in the section "Create Additional Virtual Terminal Lines" in the Configuring Terminal Lines and Modem Support chapter of the *Access and Communication Servers Configuration Guide*.

Examples

To configure virtual terminal line parameters, you would enter the following:

```
cs(config)# line vty line-number [ending-line-number]
```

Configuring the console port or virtual terminal lines allows you to perform such tasks as setting communication parameters, specifying autobaud connections, and configuring terminal operating parameters for the terminal you are using. These tasks are described later in this chapter.

The following example starts configuration for virtual terminal lines 0 to 4:

```
line vty 0 4
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

show line
show users all †

location

To record the location of a serial device, use the **location** line configuration command. Use the **no** form of this command to remove the description.

location *text*
no location

Syntax Description

text	Location description

Default

None

Command Mode

Line configuration

Usage Guidelines

The **location** command enters information about the device location and status. Use the EXEC command **show users all** to display the location information.

Example

The following example identifies the location of the console:

```
line console
location Building 3, Basement
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

show users all †

lockable

To enable the EXEC command **lock**, use the **lockable** global configuration command. Use the **no** form of this command to reinstate the default—the terminal cannot be locked.

lockable
no lockable

Syntax Description

This command has no arguments or keywords.

Default

Not lockable

Command Mode

Global configuration

Usage Guidelines

This command allows a terminal to be temporarily inaccessible by use of a temporary password.

Example

The following example sets the terminal to the lockable state:

```
lockable
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

lock †

login (line configuration)

To enable password checking at login, use the **login** line configuration command. Use the **no** form of this command to disable password checking and allow connections without a password.

login [**local** | **tacacs**]
no login

Syntax Description

local	(Optional) Selects local password checking. Authentication is based on the username specified with the **username** global configuration command.
tacacs	(Optional) Selects the TACACS-style user ID and password-checking mechanism.

Default

By default, virtual terminals require a password. If you do not set a password for a virtual terminal, it will respond to attempted connections by displaying an error message and closing the connection.

Command Mode

Line configuration

Usage Guidelines

If you specify **login** without the **local** or **tacacs** option, authentication is based on the password specified with the **password** line configuration command.

Examples

The following example sets the password *letmein* on virtual terminal line 4:

```
line vty 4
password letmein
login
```

The following example illustrates how to enable the TACACS-style user ID and password-checking mechanism:

```
line 0
password mypassword
login tacacs
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

enable password †
password
username †

login-string

To define a string of characters that the communication server sends to a host after a successful Telnet connection, use the **login-string** global configuration command. Use the **no** form of this command to remove the login string.

login-string *hostname d message* [**%***sec***p**] [**%***sec***w**] [**%b**] [**%m**] *d*
no login-string *hostname*

Syntax Description

hostname	Specifies the name of the host.
d	Sets a delimiting character of your choice—a pound sign (#) for example. You cannot use the delimiting character in the busy message.
message	Specifies the login string.
%*sec***p**	(Optional) Sets a pause in seconds. To insert pauses into the login string, embed a percent sign (%) followed by the number of seconds to pause and the letter “p.”
%*sec***w**	(Optional) Prevents users from issuing commands or keystrokes during a pause.
%b	(Optional) Sends a Break character.
%m	(Optional) Supports TN3270 terminals. Sends only CR and no LINE FEED.

Default

None

Command Mode

Global configuration

Usage Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character. To use a percent sign in the login string, precede it with another percent sign; that is, type the characters “%%.” The options can be used anywhere within the message string.

This command applies only to rlogin and Telnet sessions.

Example

In the following example, the value *%5p* causes a 5-second pause:

```
login-string office #ATDT 555-1234
%5p hello
#
```

modem answer-timeout

To set the amount of time that the communication server waits for CTS after raising DTR in response to RING, use the **modem answer-timeout** line configuration command. Use the **no** form of this command to revert to the default value for the communication server.

modem answer-timeout *seconds*
no modem answer-timeout

Syntax Description

seconds	Specifies the timeout interval in seconds.

Default

15 seconds

Command Mode

Line configuration

Usage Guidelines

This command is useful for modems that take a long time to synchronize to the appropriate line speed.

Example

The following example sets the timeout interval to 20 seconds for the modem connected to lines 3 through 13:

```
line 3 13
modem answer-timeout 20
```

Related Commands

modem callin
modem in-out

modem callin

To support dial-in modems that use DTR to control the off-hook status of the modem, use the **modem callin** line configuration command. Use the **no** form of this command to disable this feature.

modem callin
no modem callin

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

Usage Guidelines

In response to RING, the modem raises the DTR signal, which answers the modem. At the end of the session, the communication server lowers DTR, which disconnects the modem.

Example

The following example configures lines 10 through 16 for dial-in modems that can run at speeds from 300 to 19,200 bps:

```
line 10 16
modem callin
autobaud
```

Related Commands

modem answer-timeout
modem in-out

modem callout

To configure a line for reverse connections, use the **modem callout** line configuration command. Use the **no** form of this command to disable this feature.

modem callout
no modem callout

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

Example

The following example configures sets lines 17 through 32 in reverse connection mode to a large terminal switch. By using Telnet to connect to a TCP port on this host, the user gets the next free line in the rotary group.

```
line 17 32
rotary 1
modem callout
```

Related Commands

modem in-out
rotary

modem cts-required

To configure a line to require a Clear To Send (CTS) signal, use the **modem cts-required** line configuration command. Use the **no** form of this command to disable this feature.

modem cts-required
no modem cts-required

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

Usage Guidelines

This command supports lines that either the user or the network can activate. It is useful for closing connections from a user's terminal when the terminal is turned off and for preventing disabled printers and other devices in a rotary group from being considered.

Example

The following example configures a line to require a CTS signal:

```
line 5
modem cts-required
```

Related Command

rotary

modem dtr-active

To configure a line to leave DTR low unless the line has an active incoming connection or an EXEC process, use the **modem dtr-active** line configuration command. Use the **no** form of this command to disable this feature.

modem dtr-active
no modem dtr-active

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

Usage Guidelines

This command can be useful if the line is connected to an external device (for example, a timesharing system) that needs to know whether a line is in active use. The **modem dtr-active** command is similar to the **no modem** line configuration command.

Example

The following example illustrates how to configure a line for low DTR:

```
line 5
modem dtr-active
```

modem in-out

To configure a line for both incoming and outgoing calls, use the **modem in-out** line configuration command. Use the **no** form of this command to disable this feature.

modem in-out
no modem in-out

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

Usage Guidelines

The communication server does not support any dialing protocols; therefore, the host system software or the user must provide any special dialing commands when using the modem for outgoing calls.

Example

The following example illustrates how to configure a line for both incoming and outgoing calls:

```
line 5
modem in-out
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

dialer †
parity

modem ri-is-cd

To configure a line for a high-speed modem, use the **modem ri-is-cd** line configuration command. Use the **no** form of this command to disable this feature.

modem ri-is-cd
no modem ri-is-cd

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

Usage Guidelines

This command supports modems that can automatically handle telephone line activity, such as answering the telephone after a certain number of rings.

Example

The following example illustrates how to configure a line for a high-speed modem:

```
line 5
modem ri-is-cd
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

dialer †
parity

notify

To enable terminal notification about pending output from other connections, use the **notify** line configuration command. Use the **no** form of this command to end notification.

notify
no notify

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Line configuration

Usage Guidelines

This command sets a line to inform a user who has multiple, concurrent Telnet connections when output is pending on a connection other than the current one.

Example

The following example sets up notification of pending output from connections on virtual terminal lines 0 to 4:

```
line vty 0 4
notify
```

Related Command

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal notify ††

padding

To set the padding on a specific output character, use the **padding** line configuration command. Use the **no** form of this command to remove padding for the specified output character.

padding *ascii-number count*
no padding *ascii-number*

Syntax Description

ascii-number	Decimal representation of the character
count	Number of NULL bytes sent after that character, up to 255 padding characters in length

Default

None

Command Mode

Line configuration

Usage Guidelines

Use this command when the device attached is an old terminal that requires padding after certain characters (such as ones that scrolled or moved the carriage). See the “ASCII Character Set” appendix for a list of ASCII characters.

Example

The following example pads a Return (decimal 13) with 25 NULL bytes:

```
line console
padding 13 25
```

Related Command

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal padding ††

parity

To define generation of a parity bit, use the **parity** line configuration command.

parity {**none** | **even** | **odd** | **space** | **mark**}

Syntax Description

none	No parity
even	Even parity
odd	Odd parity
space	Space parity
mark	Mark parity

Default

No parity

Command Mode

Line configuration

Example

The following example changes the default of no parity to even parity:

```
line 34
parity even
```

Related Command

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal parity ††

password

To specify a password on a line, use the **password** line configuration command. Use the **no** form of this command to remove the password.

password *password*
no password

Syntax Description

password	Character string that specifies the line password. The first character cannot be a number. The string can contain any alphanumeric characters, including spaces, up to 80 characters. You cannot specify the *password* in the format *number-space-anything*. The space after the number causes problems. For example, *hello 21* is a legal password, but *21 hello* is not. The password checking is case sensitive. For example, the password *Secret* is different than the password *secret*.

Default

No password is specified.

Command Mode

Line configuration

Usage Guidelines

When an EXEC is started on a line with password protection, the EXEC prompts for the password. If the user enters the correct password, the EXEC prints its normal privileged prompt. The user can try three times to enter a password before the EXEC exits and returns the terminal to the idle state.

Example

The following example removes the password from virtual terminal lines 1 to 4:

```
line vty 1 4
no password
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

enable password †
login (line configuration)

private

To save user EXEC command changes between terminal sessions, use the **private** line configuration command. Use the **no** form of this command to restore the default condition.

private
no private

Syntax Description

This command has no arguments or keywords.

Default

User-set configuration options are cleared with the EXEC command **exit** or when the interval set with the **exec-timeout** line configuration command has passed.

Command Mode

Line configuration

Usage Guidelines

This command ensures that the terminal parameters the user sets remain in effect between terminal sessions. This behavior is desirable for terminals in private offices.

Example

The following example sets up virtual terminal line 1 to keep all user-supplied settings at system restarts:

```
line 15
private
```

Related Commands

exec-timeout
exit

refuse-message

To define a line-in-use message, use the **refuse-message** line configuration command. Use the **no** form of this command to disable the message.

refuse-message *d message d*
no refuse-message

Syntax Description

d	Delimiting character of your choice—a pound sign (#) for example. You cannot use the delimiting character in the message.
message	Message text.

Default

No line-in-use message is defined.

Command Mode

Line configuration

Usage Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character. You cannot use the delimiting character within the text of the message.

When you define a message using this command, the communication server does the following:

1 Accepts the connection.

2 Prints the custom message.

3 Clears the connection.

Example

In the following example, line 5 is configured with a line-in-use message, and the user is instructed to try again later:

```
line 5
refuse-message  /The dial-out modem is currently in use.

Please try again later./
```

rotary

To define a group of lines consisting of one of more lines, use the **rotary** line configuration command. Use the **no** form of this command to remove a line or group of lines from a rotary group.

rotary *group*
no rotary

Syntax Description

group	Integer between 1 and 100 that you choose to identify the rotary group

Default

None

Command Mode

Line configuration

Usage Guidelines

Typically, rotary groups are used on devices with multiple modem connections to allow connections to the next free line in a hunt group.

Connections to a rotary group can take advantage of the following features:

- Clear To Send (CTS)—If a line in a rotary group is configured to require CTS, the communication server skips that line if CTS from the attached device is low. This feature enables the communication server to avoid inactive host ports automatically. To enable this feature, use the **modem cts-required** line configuration command.
- RS-232 handshaking—Rotary groups are often associated with large terminal switches that require an RS-232 handshake before forming a connection. In this case, use the **modem callout** line configuration command to configure the lines in the group. If the RS-232 handshake fails on a line, the communication server steps to the next free line in the rotary group and restarts the negotiation.
- Access control—You can use access lists for groups of virtual terminal lines.
- Session timeout—Use the **session-timeout** line configuration command to set an interval for a line so that if no activity occurs on a remotely initiated connection for that interval the communication server closes the connection. The communication server assumes that the host has crashed or is otherwise inaccessible.

The remote host must specify a particular TCP port on the communication server to connect to a rotary group with connections to an individual line. The available services are the same, but the TCP port numbers are different. Table 4-3 lists the services and port numbers for both rotary groups and individual lines.

For example, if Telnet protocols are required, the remote host connects to the TCP port numbered 3000 (decimal) plus the rotary group number. If the rotary group identifier is 13, the corresponding TCP port is 3013.

Table 4-3 Services and Port Numbers for Rotary Groups and Lines

Services Provided	Base TCP Port for Rotaries	Base TCP Port for Individual Lines
Telnet protocol	3000	2000
Raw TCP protocol (no Telnet protocol)	5000	4000
Telnet protocol, binary mode	7000	6000
XRemote protocol	10000	9000

If a raw TCP stream is required, the port is 5000 (decimal) plus the rotary group number. If rotary group 5 includes a raw TCP (printer) line, the user connects to port 5005 and is connected to one of the raw printers in the group.

If Telnet binary mode is required, the port is 7000 (decimal) plus the rotary group number.

Example

The following example establishes a rotary group consisting of virtual terminal lines 2 through 4 and defines a password on those lines. By using Telnet to connect to TCP port 3001, the user gets the next free line in the rotary group. The user does not have to remember the range of line numbers associated with the password.

```
line vty 2 4
rotary 1
password letmein
login
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

dialer †
modem callout
modem cts-required
session-timeout

rxspeed

To set the terminal baud rate receive (from terminal) speed, use the **rxspeed** line configuration command.

rxspeed *bps*

Syntax Description

bps	Baud rate in bits per second (bps); see Table 4-4 for settings.

Default

9600 bps

Command Mode

Line configuration

Usage Guidelines

Set the speed to match the baud rate of whatever device you have connected to the port. Some baud rates available on devices connected to the port might not be supported on the system. The system will indicate if the speed you select is not supported. Use Table 4-4 as a guide for setting the line speeds.

Table 4-4 Router Line Speeds in Bits per Second

Communication Server Model	Baud Rates
500-CS	Any speed between 50 and 38400
ASM-CS standard	75, 110, 134, 150, 300, 600, 1200, 1800, 2000, 2400, 4800, 9600, 19200, 38400
ASM-CS nonstandard	57600, 38400, 28800, 23040, 16457, 14400, 12800, and 11520
Cisco 2500	Any speed from 50 to 115200

Example

The following example sets the line 5 receive rate to 2400 bps:

```
line 5
rxspeed 2400
```

Related Commands

speed
txspeed

script activation

To specify that a chat script start on a line any time the line is activated, use the **script activation** line configuration command. Use the **no** form of this command to disable this feature.

script activation *regexp*
no script activation

Syntax Description

regexp	Regular expression that specifies the set of modem scripts that might be executed. The first script name that matches the argument *regexp* will be used.

Default

Not assigned to terminal lines

Command Mode

Line configuration

Usage Guidelines

This command provides an asynchronous handshake to a user or device that activates the line. It can be activated by several events, such as a user issuing a carriage return on a vacant line, by a modem on the line sensing an incoming carrier, or an asynchronous device (such as another communication server) sending data. Each time an EXEC session is started on a line, the system checks to see if a **script activation** command is configured on the line. If so, and the argument *regexp* (a regular expression) matches an existing chat script name, the matched script is run on the line. For more information about regular expressions, refer to the appendix “Regular Expressions” later in this publication.

The **script activation** command can mimick a login handshake of another system. For example, a system that dials into a line on a communication server and expects an IBM mainframe login handshake can be satisfied with an appropriate activation script.

This command can also send strings to asynchronous devices that are connecting or dialing into a communication server.

The **script activation** command functions only on physical terminal (tty) lines. It does not function on virtual terminal (vty) lines.

Example

The following example specifies that the chat script with a name that includes *telebit* will be activated whenever line 4 is activated:

```
line 4
script activation telebit
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

chat-script
dialer map modem-script system-script†
dialer map modem-script system-script name†
script activation
script connection
script dialer†
script reset
script startup
start-chat

script connection

To specify that a chat script start on a line any time a remote network connection is made to a line, use the **script connection** line configuration command. Use the **no** form of this command to disable this feature.

script connection *regexp*
no script connection

Syntax Description

regexp	Specifies the set of modem scripts that might be executed. The first script name that matches the argument *regexp* will be used.

Default

Not assigned to terminal lines

Command Mode

Line configuration

Usage Guidelines

This command provides modem dialing commands and commands for logging onto remote systems. The **script connection** command functions only on physical terminal (tty) lines. It does not function on virtual terminal (vty) lines.

This command can be used to initialize an asynchronous device sitting on a line to which a reverse network connection is made.

For information about regular expressions, refer to the appendix "Regular Expressions" later in this publication.

Example

The following example specifies that the chat script with a name that includes *inband* will be activated whenever a remote connection to line 4 is established. The router can send a login string and password to the UNIX server when a network tunneling connection comes into line 4:

```
line 4
script connection inband
```

Using this example and the topology in Figure 4-1, the communication server can send a login string and password to the UNIX server when a network tunneling connection comes into line 4.

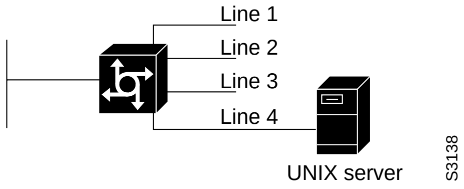

Figure 4-1 Network Tunneling Connection on an Asynchronous Line

Related Commands

A dagger (†) indicates that the command is documented in another chapter

chat-script
dialer map modem-script system-script†
dialer map modem-script system-script name†
script activation
script dialer†
script reset
script startup
start-chat

script reset

To specify that a chat script start on a line any time the specified line is reset, use the **script reset** line configuration command. Use the **no** form of this command to disable this feature.

script reset *regexp*
no script reset

Syntax Description

regexp	Specifies the set of modem scripts that might be executed. The first script name that matches the argument *regexp* will be used.

Default

Not assigned to terminal lines.

Command Mode

Line configuration

Usage Guidelines

Chat scripts provide modem dialing commands and commands for logging onto remote systems. Use this command to reset a modem attached to a line every time a call is dropped.

The **script reset** command functions only on physical terminal (tty) lines. It does not function on virtual terminal (vty) lines.

For information about regular expressions, refer to the appendix “Regular Expressions” later in this publication.

Examples

The following example specifies that any chat script name with the word *linebackup* in it will be activated any time line 7 is reset:

```
line 7
script reset linebackup
```

The following example resets a modem sitting on a line each time a call is dropped:

```
chat-script drop-line ""+++"" " " ATH OK "ATS0=1" OK "ATS9=21"
line 4
script reset drop-line
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

chat-script
dialer map modem-script system-script†
dialer map modem-script system-script name†
script activation
script connection
script dialer†
script startup
start-chat

script startup

To specify that a chat script start on a line any time the communication server is powered up, use the **script startup** line configuration command. Use the **no** form of this command to disable this feature.

script startup *regexp*
no script startup

Syntax Description

regexp	Specifies the set of modem scripts that might be executed. The first script that matches the argument *regexp* will be used.

Default

Not assigned to terminal lines

Command Mode

Line configuration

Usage Guidelines

Use this command to initialize asynchronous devices connected to a line when the communication server is powered up or reloaded. You can also use it to start up a banner other than the default banner on lines. The **script startup** command functions only on physical terminal (tty) lines. It does not function on virtual terminal (vty) lines.

For information about regular expressions, refer to the appendix “Regular Expressions” later in this publication.

Example

The following example specifies that a chat script with the word *linestart* in its name will be activated whenever line 5 is powered up:

```
line 5
script startup linestart
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

chat-script
dialer map modem-script system-script†
dialer map modem-script system-script name†
script activation
script connection
script dialer†
script reset
start-chat

service linenumber

To configure the communication server to display line number information after the EXEC or incoming banner, use the **service linenumber** global configuration command. Use the **no** form of this command to disable this function.

service linenumber
no service linenumber

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

With the **service linenumber** command, you can have the communication server display the host name, line number, and location each time an EXEC is started or an incoming connection is made. The line number banner appears immediately after the EXEC banner or incoming banner. It is useful for tracking problems with modems because the host and line for the modem connection are listed. Modem type information can also be included.

Example

The following example illustrates the type of line number information that can appear after the EXEC banner:

```
user1@location1%telnet cs2 2001
Trying 131.109.44.37
Connected to user1-gw.cisco.com
Escape character is '^]'.

cs1 line 1 virtual terminal 0
```

session-limit

To set the maximum number of terminal sessions per line, use the **session-limit** line configuration command. Use the **no** form of this command to remove any specified session limit.

session-limit *session-number*
no session-limit

Syntax Description

session-number	Specifies the maximum number of sessions.

Default

The default and set session limits are displayed with the **show terminal** EXEC command.

Command Mode

Line configuration

Example

The following example limits the number of sessions to eight on a ten-line range:

```
line 2 12
session-limit 8
```

session-timeout

To set the interval for closing the connection when there is no input or output traffic, use the **session-timeout** line configuration command. Use the **no** form of this command to remove the timeout definition.

session-timeout *minutes* [**output**]
no session-timeout

Syntax Description

minutes Specifies the timeout interval in minutes.

output (Optional) Specifies that when traffic is sent to an asynchronous line from the communication server (within the specified interval), the connection is retained.

Default

The default interval is zero, indicating the communication server maintains the connection indefinitely.

Command Mode

Line configuration

Usage Guidelines

This command sets the interval that the communication server waits for traffic before closing the connection to a remote computer and returning the terminal to an idle state. If the keyword **output** is not specified, the session timeout interval is based solely on detected input from the user. You can specify a session timeout on each port.

Examples

The following example sets an interval of 20 minutes and specifies that the timeout is subject to traffic detected from the user (input only):

```
line 5
session-timeout 20
```

The following example sets an interval of 10 minutes, subject to traffic on the line in either direction:

```
line 5
session-timeout 10 output
```

show line

To display a terminal line’s parameters, use the **show line** EXEC command.

show line [*line-number*]

Syntax Description

line-number (Optional) Absolute line number of the line for which you want to list parameters

Command Mode

EXEC

Sample Display

The following sample output from the **show line** command shows that line 2 is a virtual terminal line with a transmit and receive rate of 9600 bps. Also shown is the modem state, terminal screen width and length, and so on.

```
cs# show line 2

 Tty Typ    Tx/Rx    A Modem  Roty AccO AccI  Uses    Noise   Overruns
   2 VTY  9600/9600  -    -      -    -    -     0        0        0/0

Line 2, Location: "", Type: ""
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600/9600
Status: No Exit Banner
Capabilities: none
Modem state: Idle
Special Chars: Escape  Hold  Stop  Start  Disconnect  Activation
                ^^x    none   -     -       none
Timeouts:      Idle EXEC    Idle Session   Modem Answer  Session   Dispatch
               0:10:00       never         0:00:15      not imp   not set
Session limit is not set.
Editing is enabled.
History is enabled, history size is 10.
Allowed transports are telnet mop.  Preferred is telnet.
No output characters are padded
Characters causing immediate data dispatching:
   Char    ASCII
```

show terminal

Use the **show terminal** EXEC command to obtain information about the terminal configuration parameter settings for the current terminal line.

show terminal

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show terminal** command:

```
cs# show terminal

Line 2, Location: "", Type: ""
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600/9600
Status: Ready, Active, No Exit Banner
Capabilities: Enabled
Modem state: Ready
Special Chars: Escape  Hold  Stop  Start  Disconnect  Activation
                ^^x    none   -     -       none
Timeouts:      Idle EXEC    Idle Session   Modem Answer  Session   Dispatch
                never         never         0:00:15      not imp   not set
Session limit is not set.
Allowed transports are telnet rlogin.  Preferred is telnet
No output characters are padded
```

Table 4-5 describes the fields in the first three lines of **show terminal** output.

Table 4-5 Show Terminal Field Descriptions—First Three Lines of Output

Field	Description
Line 2	Current terminal line
Location:""	Location of the current terminal line, as specified using the **location** line configuration command
Type: ""	Type of the current terminal line, as specified using the **line** global configuration command
Length: 24 lines	Length of the terminal display
Width: 80 columns	Width of the terminal display, in character columns
Baud rate (TX/RX) is 9600/9600	Transmit rate/receive rate of the line

The following line of output indicates the status of the line:

```
Status: Ready, Active, No Exit Banner
```

Table 4-6 describes possible values for the Status field.

Table 4-6 Show Terminal Field Description—Status Field

Field	Description
Active	A process is actively using the line.
Autobauding	The line is running the autobaud process.
Carrier Dropped	Some sense of “carrier” has been dropped, so the line process should be killed.
Connected	The line has at least one active connection.
Dialing Out	A DDR asynchronous interface is dialing a remote site on this line.
Echo Off	The line is not echoing what the user types in (because a password must be entered, for example).
Escape Started	The first character of the escape sequence has been typed.
Escape Typed	Both characters of the escape sequence have been typed.
Hanging Up	The line state is “hanging up.”
Hardware XON/XOFF	The line uses a UART that supports XON/XOFF flow control in hardware. (This does not mean that the line is currently using software flow control.)
Hold Typed	The user typed the “hold character” (and the line is paused).
Idle	The line modem state is “idle” (see modem state diagrams).
Idle Timeout	An idle timeout has occurred.
Input Stopped	The input has been turned off due to hardware flow control or overflow.
No Exit Banner	The normal exit banner will not be displayed on this line.
PSI Enabled	The line is paying attention to typed escape characters.
Rcvd BREAK	A BREAK sequence has been received on the line.
Rcvd Command	The line has received a special command sequence (four example, ^^B for send break).
Rcvd CR	The last character received was a carriage return.
Ready	The line state is “ready.”
Ring Transition	There has been a transition on the RING signal of the line.
Send Break Soon	You need to send a BREAK on the line soon.
Send XOFF Soon	Your buffers are full and you should send an XOFF soon.
Sending Break	You are in the process of sending a BREAK sequence on the line.
Sent XOFF	Your buffers were full, so we sent an XOFF.
SLIP Mode	The line is running SLIP or PPP.

The following line of output indicates the status of the capabilities of the line; these capabilities correspond closely to configurable parameters that can be set using configuration commands.

```
Capabilities: Enabled
```

Table 4-7 describes possible values for the Capabilities field.

Table 4-7 Show Terminal Field Descriptions—Capabilities Field

Field	Description
Autobaud Full Range	Corresponds to the **autobaud** command.
Character Padding	At least one **pad** *c x* configuration command has been used.
Enabled	The user has "enabled" successfully.
EXEC Suppressed	Corresponds to the **no exec** command.
Hangup on Last Close	Corresponds to the **autohangup** command.
Hardware Flowcontrol In	Corresponds to the **flowcontrol hardware in** command.
Hardware Flowcontrol Out	Corresponds to the **flowcontrol hardware out** command.
Insecure	Corresponds to the **insecure** command.
Lockable	Corresponds to the **lockable** command.
Modem Callin	Corresponds to the **modem callin** command.
Modem Callout	Corresponds to the **modem callout** command.
Modem CTS-Required	Corresponds to the **modem cts-required** command.
Modem DTR-Active	Corresponds to the **modem dtr-active** command.
Modem RI is CD	Corresponds to the **modem ri-is-cd** command.
No Login Banner	Corresponds to the **no exec-banner** command.
Notification Set	Corresponds to the **notify** command.
Output Non-Idle	Corresponds to the **session-timeout** command.
Permanent SLIP	Corresponds to the **slip-dedicated** command
Private Line	Corresponds to the **private** command.
Refuse Suppress-GA	Corresponds to the **telnet refuse** command.
Receives Logging Output	Corresponds to the **monitor** command.
Refuse Telnet Echo	Corresponds to the **telnet refuse** command.
Send BREAK on IP	Corresponds to the **telnet break-on-ip** command.
SLIP allowed	Corresponds to the **slip address** command.
Software Flowcontrol In	Corresponds to the **flowcontrol software in** command.
Software Flowcontrol Out	Corresponds to the **flowcontrol software out** command.
Telnet Transparent Mode	Corresponds to the **telnet transparent** command.

The following line of output indicates the modem state. Possible values include Autobauding, Carrier Dropped, Hanging Up, Idle, and Ready.

```
Modem state: Ready
```

The following lines of output indicate the special characters that can be entered to activate various terminal operations. The none or hyphen (-) values imply that no special characters are set.

```
Special Chars: Escape  Hold  Stop  Start  Disconnect  Activation
                ^^x    none   -     -       none
```

The following lines of output indicate the timeout values that have been configured for the line:

```
Timeouts:      Idle EXEC    Idle Session   Modem Answer  Session   Dispatch
                never         never         0:00:15      not imp   not set
```

Table 4-8 describes the fields in the preceding lines of output.

Table 4-8 Show Terminal Field Descriptions—Timeouts Fields

Field	Description
Idle EXEC	Interval that the EXEC command interpreter waits for user input before resuming the current connection; or if no connections exist, returning the terminal to the idle state and disconnecting the incoming session. This interval is set using the **exec-timeout** command.
Idle Session	Interval that the communication server waits for traffic before closing the connection to a remote computer and returning the terminal to an idle state. This interval is set using the **session-timeout** command.
Modem Answer	Interval during which the communication server raises DTR in response to RING and the modem response to CTS. This interval is set using the **modem answer-timeout** command.
Session	Not implemented in this release.
Dispatch	Number of milliseconds the communication server waits after putting the first character into a packet buffer before sending the packet. This interval is set using the **dispatch-timeout** command.

The following lines of output indicate how various options have been configured:

```
Session limit is not set.
Allowed transports are telnet rlogin. Preferred is telnet
No output characters are padded
```

special-character-bits

To configure the number of data bits per character for special characters such as software flow control characters and escape characters, use the **special-character-bits** line configuration command.

special-character-bits {**7** | **8**}

Syntax Description

7	Selects the 7-bit ASCII character set.
8	Selects the full 8-bit character set for special characters.

Default

7-bit ASCII character set

Command Mode

Line configuration

Usage Guidelines

Setting the special character bits to 8 allows you to use twice as many special characters as with the 7-bit ASCII character set. The special characters affected by this setting are the escape, hold, stop, start, disconnect, and activation characters.

Example

The following example allows the full 8-bit international character set for special characters on line 5:

```
line 5
special-character-bits 8
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

default-value exec-character-bits
default-value special-character-bits
exec-character-bits
terminal exec-character-bits †
terminal special-character-bits †

speed

To set the terminal baud rate, use the **speed** line configuration command. This command sets both the transmit (to terminal) and receive (from terminal) speeds.

speed *bps*

Syntax Description

bps	Baud rate in bits per second (bps); see Table 4-9 for settings.

Default

9600 bps

Command Mode

Line configuration

Usage Guidelines

Set the speed to match the baud rate of whatever device you have connected to the port. Some baud rates available on devices connected to the port might not be supported on the communication server. The communication server indicates whether the speed you selected is not supported. Use Table 4-9 as a guide for setting the line speeds.

Table 4-9 Communication Server Line Speeds in Bits per Second

Communication Server Model	Baud Rates
500-CS	Any speed between 50 and 38400
ASM-CS standard	75, 110, 134, 150, 300, 600, 1200, 1800, 2000, 2400, 4800, 9600, 19200, 38400
ASM-CS nonstandard	57600, 38400, 28800, 23040, 16457, 14400, 12800, and 11520
Cisco 2500	Any speed from 50 to 115200

Example

The following example sets lines 1 and 2, which are paired on an ASM serial line card, to a nonstandard 38,400 bps:

```
line 1 2
speed 38400
```

Related Commands

rxspeed
txspeed

start-character

To set the flow control start character, use the **start-character** line configuration command. Use the **no** form of this command to remove the character.

start-character *ascii-number*
no start-character

Syntax Description

ascii-number	Decimal representation of the start character

Default

Decimal 17

Command Mode

Line configuration

Usage Guidelines

This command defines the character that signals the start of data transmission when software flow control is in effect. See the "ASCII Character Set" appendix for a list of ASCII characters.

Example

The following example changes the start character to Ctrl-B, which is decimal 2:

```
line 2
start-character 2
```

Related Commands

flowcontrol
stop-character

start-chat

To specify that a chat script start on a specified line at any point, use the **start-chat** privileged EXEC command. Use the **no** form of this command to stop the chat script.

start-chat *regexp* [*line-number* [*dialer-string*]]
no start-chat

Syntax Description

regexp	Specifies the name of a regular expression or modem script to be executed. If there is more than one script with a name that matches the argument *regexp*, the first script found will be used.
line-number	(Optional) Indicates the line number on which to execute the chat script. If you do not specify a line number, the current line number is chosen. If the specified line is busy, the script is not executed and an error message appears. If the *dialer-string* argument is specified, *line-number* must be entered; it is not optional if you specify a dialer string.
dialer-string	(Optional) String of characters (often a telephone number) to be sent to a DCE. If you enter a dialer string, you must also specify *line-number*, or the chat script *regexp* will not start.

Command Mode

Privileged EXEC

Usage Guidelines

This command provides modem dialing commands for a chat script that you want to apply immediately to a line. If you do not specify a line, the script runs on the current line. If the specified line is already in use, the script is not activated and an error message appears.

The argument *regexp* is used to specify the name of the modem script that is to be executed. The first script that matches the argument in this command and the **dialer map** command will be used. For more information about regular expressions, refer to the “Regular Expressions” appendix later in this publication.

This command functions only on physical terminal (tty) lines. It does not function on virtual terminal (vty) lines.

Example

The following example forces a dialout on line 8 using the script *telebit*:

```
Cisco2509# start-chat telebit line 8
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

chat-script
dialer map modem-script system-script†
dialer map modem-script system-script name†

script activation
script connection
script dialer[†]
script reset
script startup

state-machine

To specify the transition criteria for the state of a particular state machine, use the **state-machine** global configuration command.

state-machine *name state firstchar… lastchar* [*nextstate* | **transmit**]

Syntax Description

name	Specifies the name for the state machine (used in the **dispatch-machine** line command). The user can specify any number of state machines, but each line can have only one state machine associated with it.
state	Defines which state is being modified. There are a maximum of eight states per state machine. Lines are initialized to state 0 and return to state 0 after a packet is transmitted.
firstchar… lastchar	Specify a range of characters. If the state machine is in the indicated state, and the next character input is within this range, the process goes to the specified next state. Full 8-bit character comparisons are done, so the maximum value is 255. Take care that the line is configured to strip parity bits (or not generate them) or duplicate the low characters in the upper half of the space.
nextstate	(Optional.) Defines the state to enter if the character is in the specified range.
transmit	(Optional) Causes the packet to be transmitted and the state machine to be reset to state 0. Recurring characters that have not been explicitly defined to have a particular action return the state machine to state 0.

Default

No transition criteria are specified.

Command Mode

Global configuration

Usage Guidelines

This command is paired with the **dispatch-machine** line configuration command, which defines the line on which the state machine is effective.

Example

The following example uses a dispatch machine named *function* to ensure that the function key characters on an ANSI terminal are lumped together in one packet. Because the default in the example is to remain in state 0 without transmitting anything, normal key signals are transmitted immediately:

```
line 1 20
dispatch-machine function
!
state-machine function 0 0 255 transmit
```

Related Commands

dispatch-character
dispatch-machine
dispatch-timeout

stop-character

To set the flow control stop character, use the **stop-character** line configuration command. Use the **no** form of this command to remove the character.

stop-character *ascii-number*
no stop-character

Syntax Description

ascii-number	Decimal representation of the stop character

Default

Decimal 19

Command Mode

Line configuration

Usage Guidelines

This command defines the character that signals the end of data transmission when software flow control is in effect. See the "ASCII Character Set" appendix for a list of ASCII characters.

Example

The following example changes the stop character to Ctrl-E, which is decimal 5:

```
line 3
stop-character 5
```

Related Commands

flowcontrol
start-character

stopbits

To set the number of the stop bits transmitted per byte, use the **stopbits** line configuration command.

stopbits {**1** | **1.5** | **2**}

Syntax Description

1	One stop bit
1.5	One and one-half stop bits
2	Two stop bits

Default

2 stop bits

Command Mode

Line configuration

Example

The following example changes the default from 2 stop bits to 1 as a performance enhancement:

```
line 4
stopbits 1
```

terminal-type

To specify the type of terminal connected to a line, use the **terminal-type** line configuration command. Use the **no** form of this command to remove any information about the type of terminal and reset the line to the default terminal emulation.

terminal-type {*terminal-name* | *terminal-type*}
no terminal-type

Syntax Description

terminal-name	Terminal name
terminal-type	Terminal type

Default

VT100

Command Mode

Line configuration

Usage Guidelines

This command records the type of terminal connected to the line. The argument *terminal-name* provides a record of the terminal type and allows terminal negotiation of display management by hosts that provide that type of service.

For TN3270 applications, this command must follow the corresponding ttycap entry in the configuration file.

Example

The following example defines the terminal on line 7 as a VT220:

```
line 7
terminal-type VT220
```

Related Command

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal terminal-type ††

transport input

To allow the system administrator to define which protocols to use to connect to a specific line of the access server, use the **transport input** line configuration command.

transport input {**lat** | **mop** | **none** | **pad** | **rlogin** | **telnet** | **all**}

Syntax Description

input	Defines which protocols to use to on this line when connecting on an incoming call.
lat	Selects the Digital LAT protocol and specifies both incoming reverse LAT and host-initiated connections.
mop	Selects the MOP protocol.
none	Prevents any protocol selection on the line. This makes the port unusable by incoming connections.
pad	Selects X.3 PAD incoming connections.
rlogin	Selects the UNIX rlogin protocol.
telnet	Specifies all types of incoming TCP/IP connections.
all	Selects all protocols. Restores the default configuration.

Default

All protocols allowed on the line

Command Mode

Line configuration

Usage Guidelines

You can specify one protocol, multiple protocols, all protocols, or no protocols. To specify multiple protocols, type the keyword for each protocol, separated by a space.

This command can be useful in distributing resources among different types of users, or making certain that only specific hosts can access a particular port. When using protocol translation, the **transport input** command is useful in controlling exactly which protocols can be translated to other protocols when using two-step translation.

Access lists for each individual protocol can be defined in addition to the allowances created by the **transport input** command. Any settings made with the **transport input** command override settings made with the **transport preferred** command.

Note Unlike defaults for other commands, the default **transport input all** displays when you perform **show running config** on TTY and AUX lines to display current settings.

Example

The following example sets the incoming protocol to Telnet for vtys 0 to 32:

```
line vty 0 32
transport input telnet
```

Related Commands

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal transport input ††
transport output
transport preferred

transport output

To determine the protocols that can be used for outgoing connections from a line, use the **transport output** line configuration command.

transport output {**lat** | **none** | **pad** | **rlogin** | **telnet** | **all**}

Syntax Description

lat Selects the Digital LAT protocol, which is the protocol used most often to connect access servers to Digital hosts.

none Prevents any protocol selection on the line. The system normally assumes that any unrecognized command is a host name. If the protocol is set to **none**, the system no longer makes that assumption. No connection will be attempted if the command is not recognized.

pad Selects X.3 PAD, used most often to connect access servers to X.25 hosts.

rlogin Selects the UNIX rlogin protocol for TCP connections. The rlogin setting is a special case of Telnet. If an rlogin attempt to a particular host has failed, the failure will be tracked, and subsequent connection attempts will use Telnet instead.

telnet Selects the TCP/IP Telnet protocol. It allows a user at one site to establish a TCP connection to a login server at another site.

all Selects all protocols.

Default

telnet

Command Mode

Line configuration

Usage Guidelines

You can specify one protocol, multiple protocols, all protocols, or no protocols. To specify multiple protocols, type the keyword for each protocol, separated by a space.

Any settings made with the **transport output** command override settings made with the **transport preferred** command.

Example

The following example prevents any protocol selection:

```
transport output none
```

Related Commands

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal transport output ††
transport input
transport preferred

transport preferred

To specify the transport protocol the communication server uses if the user does not specify one when initiating a connection, use the **transport preferred** line configuration command.

transport preferred {**lat** | **none** | **pad** | **rlogin** | **telnet**}

Syntax Description

lat Selects the Digital LAT protocol, which is the protocol used most often to connect communication servers to Digital hosts.

none Prevents any protocol selection on the line. The system normally assumes that any unrecognized command is a host name. If the protocol is set to **none**, the system no longer makes that assumption. No connection will be attempted if the command is not recognized.

pad Selects X.3 PAD, used most often to connect communication servers to X.25 hosts.

rlogin Selects the UNIX rlogin protocol for TCP connections. The rlogin setting is a special case of Telnet. If an rlogin attempt to a particular host has failed, the failure will be tracked, and subsequent connection attempts will use Telnet instead.

telnet Selects the TCP/IP Telnet protocol. It allows a user at one site to establish a TCP connection to a login server at another site.

Default

Telnet

Command Mode

Line configuration

Usage Guidelines

Specify **transport preferred none** to prevent errant connection attempts.

Example

The following example sets the preferred protocol to Telnet on physical terminal line 1:

```
line tty 1
transport preferred telnet
```

Related Commands

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal transport preferred ††
transport input
transport output

txspeed

To set the terminal transmit baud rate (to terminal), use the **txspeed** line configuration command.

txspeed *bps*

Syntax Description

bps	Baud rate in bits per second (bps); see Table 4-10 for settings.

Default

9600 bps

Command Mode

Line configuration

Usage Guidelines

Set the speed to match the baud rate of whatever device you have connected to the port. Some baud rates available on devices connected to the port might not be supported on the communication server. The communication server will indicate if the speed you select is not supported. Use Table 4-10 as a guide for setting the line speeds.

Table 4-10 Router Line Speeds in Bits per Second

Communication Server Model	Baud Rates
500-CS	Any speed between 50 and 38400
ASM-CS standard	75, 110, 134, 150, 300, 600, 1200, 1800, 2000, 2400, 4800, 9600, 19200, 38400
ASM-CS nonstandard	57600, 38400, 28800, 23040, 16457, 14400, 12800, and 11520
Cisco 2500	Any speed from 50 to 115200

Example

The following example sets the transmit speed for line 5 to 2400 bps:

```
line 5
txspeed 2400
```

Related Commands

rxspeed
speed

vacant-message

To display an idle terminal message, use the **vacant-message** line configuration command. Use the **no** form of this command to remove the default vacant message or any other vacant message that may have been set.

vacant-message [*d message d*]
no vacant-message

Syntax Description

d (Optional) A delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.

message (Optional) Vacant terminal message.

Default

The format of the default vacant message is as follows:

```
<blank lines>
hostname tty# is now available
<blank lines>
Press RETURN to get started.
```

This message is generated by the system.

Command Mode

Line configuration

Usage Guidelines

This command enables the banner to be displayed on the screen of an idle terminal. The **vacant-message** command without any arguments restores the default message.

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

Note For a rotary group, you only need to define the message for the first line in the group.

Example

The following example turns on the system banner and displays this message:

```
line 0
vacant-message #
                Welcome to Cisco Systems, Inc.
                 Press Return to get started.
#
```

width

To set the terminal screen width, use the **width** line configuration command. This command sets the number of character columns displayed on the attached terminal.

width *characters*

Syntax Description

characters	Integer that specifies the number of character columns displayed on the terminal.

Default

80 character columns

Command Mode

Line configuration

Usage Guidelines

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

Some hosts can learn the values for both length and width specified with the **line** and **width** commands.

Example

The following example changes the character columns to 132 for the terminal on line 7:

```
line 7
location console terminal
width 132
```

Related Command

Two daggers (††) indicate that the command is documented in the *Cisco Access Connection Guide*.

terminal width ††