CHAPTER 4

Line Configuration and Terminal Setting Commands

The line configuration commands described in this chapter are used to configure lines on the communication server. For line configuration task descriptions, refer to the *Communication Server Configuration Guide*.

The **editing** and **history size** line configuration commands and the **terminal editing** and **terminal history size** EXEC commands are described with general user interface commands in the chapter "Understanding the User Interface."

See the *Communication Server and Protocol Translator Connection Guide* for information about the EXEC commands that set terminal parameters for the duration of a session.

activation-character

To set the activation character, use the **activation-character** line command. This command defines the character you type at a vacant terminal to begin a terminal session. The **no activation-character**line configuration command causes any character to activate a terminal.

activation-character *ASCII-number*
no activation-character

Syntax Description

ASCII-number	The ASCII decimal representation of the activation character

Default

Return (ASCII character 13)

Command Mode

Line configuration

Example

The following example sets the session activation character to a Ctrl-R, which is ASCII decimal character 18:

```
line 10
activation-character 18
```

autobaud

To set the line for automatic baud detection over a range from 300 to 19200 baud, use the **autobaud** line configuration command. The **no autobaud** command restores the default.

autobaud [**fast**]
no autobaud

Syntax Description

fast	(Optional.) Identifies the baud rate with exactly three carriage returns.

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 19,200 baud when the parity bit is set because of hardware limitations.

With the **fast** option, simple startup scripts can manage logging onto serial ports on the communication server.

Example

The following example sets line 5 for autobaud detection:

```
line 5
autobaud
```

autocommand

To set it a command or list of commands to automatically execute, use the **autocommand** line configuration command. This command causes any commands associated with it to be executed.

autocommand *command*

Syntax Description

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

Default

None

Command Mode

Line configuration

Example

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

```
line vty 96
autocommand connect dustbin uucp /stream
```

autohangup

To configure automatic line disconnect, use the **autohangup** line configuration command. The 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
```

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 (line activated, or incoming connection to VTY).

banner exec *d message d*

Syntax Description

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

message The message.

Default

None

Command Mode

Global

Usage Guidelines

Follow the 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.

The **banner** global configuration commands (all versions) are only applicable in two-step protocol translation sessions, and when using the system console.

Example

The following example sets an EXEC message:

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

Related Commands

banner incoming
banner motd
exec-banner

banner incoming

To display messages on terminals connected to reverse Telnet lines, use the **banner incoming** global configuration command. An incoming connection is one initiated from the Ethernet side of the communication server. The EXEC banner can be suppressed on certain lines using the **no exec-banner** line command. This line should *not* display the EXEC or MOTD banners when an EXEC is created.

banner incoming *d message d*

Syntax Description

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

message The message.

Default

None

Command Mode

Global configuration

Usage Guidelines

Follow the 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.

The **banner** global configuration commands (all versions) are only applicable in two-step protocol translation sessions, and when using the system console.

Note Messages are never displayed on incoming stream type connections, because they might interfere with printer daemons.

Example

The following example sets an incoming connection message:

```
banner incoming #
Welcome to Rhesus. Use the enable password to access configure mode.
#
```

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

message The message.

Default

None

Command Mode

Global configuration

Usage Guidelines

Follow the 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 message-of-the-day 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 (when no keyword specified). Similarly, if a **banner** command is added to the configuration, any exiting **banner motd** command is overwritten.

The **banner** global configuration commands (all versions) are only applicable in two-step protocol translation sessions, and when using the system console.

Example

The following example uses the pound sign (#) 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 display a “host failed” message when a connection fails, use the **busy-message** global configuration command. The command defines a message that the communication server displays whenever an attempt to connect to the specified host fails. Use the **no busy-message** 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 The name of the host.

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

message The “host failed” message.

Default

None

Command Mode

Global

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 messages.

Note The global configuration command **busy-message** is only applicable in two-step protocol translation sessions, and when using the system console.

Example

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

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

Related Command

autocommand

databits

To set the number of data bits per character, use the **databits** line configuration command.

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

Syntax Description

5 Five data bits

6 Six data bits

7 Seven data bits

8 Eight data bits

Default

8 data bits per character

Command Mode

Line configuration

Usage Guidelines

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 databits from the default of 8 to 7 on line 4:

```
line 4
databits 7
```

default-value exec-character-bits

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

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

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

default-value special-character-bits

exec-character-bits

special-character-bits

terminal data-character-bits

terminal exec-character-bits

terminal special-character-bits

default-value special-character-bits

The **default-value special-character-bits** global configuration command allows you to configure the flow control default from a 7-bit width to 8 bits, thus providing for twice as many special characters to be used.

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

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 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 ASCII special character set:

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

Related Commands

default-value exec-character-bits

exec-character-bits

special-character-bits

terminal data-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. This command defines the character you enter to end a terminal session. The **no disconnect-character** command removes the disconnect character.

disconnect-character *ASCII-number*
no disconnect-character

Syntax Description

ASCII-number	The ASCII decimal representation of the session disconnect character

Default

None

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.

Example

The following example sets the session disconnect character to Ctrl-X, which is ASCII decimal character 24:

```
line 14
disconnect-character 24
```

dispatch-character

To define a character that causes a packet to be sent, use the **dispatch-character** line configuration command. This command defines the dispatch character even if the dispatch timer has not expired. The **no dispatch-character** command removes 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	The ASCII decimal representation of the character, such as Return (ASCII character 13) for line-at-a-time transmissions

Default

None

Command Mode

Line configuration

Usage Guidelines

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

Example

The following example specifies two Returns as the dispatch character:

```
dispatch-character 13 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 Specifies the name of the state machine that determines when to send packets on the asynchronous line

Default

None

Command Mode

Line command

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 if a state machine is also specified.

If a packet becomes full, it will be sent regardless of the current state. However, the state is not reset. The packet size is dependent on the amount of traffic on the asynchronous line, as well as the dispatch timeout. There is always room for 60 data bytes, and if the dispatch-timeout is 100 ms or greater, 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. The **no dispatch-timeout** command removes the timeout definition.

dispatch-timeout *milliseconds*
no dispatch-timeout *milliseconds*

Syntax Description

milliseconds An 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 can be added to the packet, thus increasing the processing efficiency of the remote host.

Default

None

Command Mode

Line configuration

Usage Guidelines

The **dispatch-character** and **dispatch-timeout** commands together cause the communication server to attempt to buffer characters into larger packets for transmission to the remote host. The communication server normally dispatches each character as it is entered.

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 1 20
dispatch-timeout 80
```

Related Commands

dispatch-character
dispatch-machine
state-machine

escape-character

To define a system escape character, use the **escape-character** line configuration command. The **no escape-character** command sets the escape character to Break. (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.)

escape-character *ASCII-number*
no escape-character

Syntax Description

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

Default

Ctrl ^

Command Mode

Line configuration

Example

The following example sets the escape character to Ctrl-Q, which is ASCII decimal character 17:

```
line console
escape-character 17
```

exec

To start an EXEC process on a line, use the **exec** line configuration command. The **no exec** command turns off the EXEC process for the line specified. A serial printer, for example, should not have an EXEC started.

exec
no exec

Syntax Description

This command has no arguments or keywords.

Default

By default, the communication server starts EXECs on all lines.

Command Mode

Line configuration

Example

The following example illustrates how to configure a line for a printer:

```
line 7
location Printer 1
no exec
speed 19200
```

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. The **no exec-banner** command suppresses the banner messages.

exec-banner
no exec-banner

Syntax Description

This commands has no arguments or keywords.

Default

By 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 asynchronous terminal lines 0 through 4:

```
line tty 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 {**8** | **7**}

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

Line configuration

Usage Guidelines

Configuring 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 global configuration command and the line configuration commands.

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

Related Commands

default-value exec-character-bits

default-value special-character-bits

special-character-bits

terminal data-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. 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. The **no exec-timeout** command removes the timeout definition. It is the same as entering **exec-timeout 0**.

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

Syntax Description

minutes Integer that specifies the number of minutes.

seconds Additional time intervals in seconds. An interval of zero specifies no time-outs.

Default

10 minutes

Command Mode

Line configuration

Examples

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

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

The following example sets an interval of 10 seconds:

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

Related Command

private

flowcontrol

To set the flow control, use the **flowcontrol** line configuration command. The command sets the method of data flow control between the terminal or other serial device and the communication server.

flowcontrol {**none** | **software** [**in** | **out**] | **hardware**}

Syntax Description

none	Turns off flow control.
software	Sets software flow control. (Optional.) 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. For more information about hardware flow control, see the hardware reference manual for your server product. Only limited modem control is possible on a line using hardware flow control. On the 500-CS, this command can be used to achieve hardware flow control to and from the 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 flowcontrol 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. The **no hold-character** command restores the default.

hold-character *ASCII-number*
no hold-character

Syntax Description

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

Default

No local hold character 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.

Example

The following example sets the hold character to Ctrl-H, which is ASCII decimal character 8:

```
line 56
hold-character 8
```

insecure

To set the line as in an insecure location, use the **insecure** line command. The **no insecure** command disables this feature.

insecure
no insecure

Syntax Description

This commands has no arguments or keywords.

Default

Disabled

Command Mode

Line command

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. The communication server software uses the value of this command to determine when to pause during multiple-screen output.

length *screen-length*

Syntax Description

screen-length	The 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

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 command collection mode, use the **line** global configuration command.

line [*type-keyword*] *line-number* [*ending-line-number*]

Syntax Description

type-keyword	Specifies the type of line to be configured; it is one of the keywords listed in Table 1-1.
line-number	Specifies 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. The communication server displays an error message if you do not specify a line number.
ending-line-number	Specifies the relative number of the last line in a contiguous group you want to configure. If you omit *type-keyword*, then *line-number* and *ending-line-number* are absolute rather than relative line numbers.

Table 1-1 Line Command Mode Keywords

Keywords	Line Command Mode
console	Console terminal line.
aux	Auxiliary line. ASM-CS only.
printer	Parallel printer line.
tty	Standard asynchronous line.
vty	A virtual terminal for remote console access. The communication server host can support five virtual terminals for access by incoming Telnet, LAT, or MOP connections.

Default

None

Command Mode

Global configuration

Usage Guidelines

You can display the absolute and relative line numbers by using the EXEC command **show users all**.

Note Line numbers are in octal on the ASM-CS communication servers, but they are in decimal form on the 500-CS communication server.

Example

The following example starts configuration for the first five asynchronous terminal lines, 0 through 4:

```
line tty 0 4
```

location

To record the location of a serial device, use the **location** line configuration command. The **no location** command removes the description.

location *text*
no location

Syntax Description

text Location description

Default

None

Command Mode

Line command

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 device on line 10:

```
line 10
location Terminal 10 Bldg 3
```

lockable

To enable the EXEC command **lock**, use the **lockable** global configuration command. This command allows a terminal to be temporarily inaccessible by use of a temporary password. The **no lockable** command reinstates the default, which does not allow the terminal to be locked.

lockable
no lockable

Syntax Description

This command has no arguments or keywords.

Default

Not lockable

Command Mode

Global configuration

Note The global configuration command **lockable** is only applicable in two-step protocol translation sessions and when using the system console.

Example

The following example sets the terminals connected to the communication server to the lockable state:

```
lockable
```

login (line configuration)

To enable password checking at login, use the **login** line configuration command. Use the **no login** 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 †

modem answer-timeout

To configure line modem timing, use the **modem answer-timeout** line configuration command. The command sets the interval during which the communication server raises DTR in response to RING and the modem responds to CTS. This behavior is useful for modems that take a long time to synchronize to the appropriate line speed. The **no modem answer-timeout** command disables the line for modem timing.

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

Syntax Description

seconds Specifies the timeout interval in seconds.

Default

15 seconds

Command Mode

Line configuration

Example

The following example specifies a 30-second synchronization period for the modem connected to lines 3 through 13:

```
line 3 13
modem answer-timeout 30
```

modem callin

To configure a line for a dial-in modem, use the **modem callin** line configuration command. The command supports dial-in modems that use DTR to control the off-hook status of the telephone line. Raising the DTR signal answers the telephone; lowering DTR hangs up the telephone. The **no modem callin** command disables the line for a dial-in modem.

modem callin
no modem callin

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

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
```

modem callout

To configure a line for reverse connections, use the **modem callout** line configuration command. The command supports ports connected to computers that are to be connected to modems. This command causes the communication server to behave somewhat like a modem. The **no modem callout** command disables the line for reverse connections.

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 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 Command

rotary

modem cts-required

To configure a line to require CTS, use the **modem cts-required** line configuration command. The command supports lines that either the user or the network can activate. This command 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. The **no modem cts-required** command disables a line to require CTS.

modem cts-required
no modem cts-required

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

Example

The following example shows how to set line 5 to require CTS:

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

modem dtr-active

To configure a line for low DTR, use the **modem dtr-active** line configuration command. The command configures a line to leave DTR low unless the line has an active incoming connection or an EXEC process. This behavior 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 command. The **no modem dtr-active** command disables a line for low DTR.

modem dtr-active
no modem dtr-active

Syntax Description

This command has no arguments or keywords.

Default

No modem control

Command Mode

Line configuration

Example

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

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

modem inout

To configure a line for both incoming and outgoing calls, use the **modem inout** line configuration command. The command enables a line to be used for both incoming and outgoing calls on dial-in/dial-out modems. Use the **no modem inout** command to disable this function.

modem inout
no modem inout

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 inout
```

modem ri-is-cd

To configure a line for a high-speed modem, use the **modem ri-is-cd** line configuration command. The command supports modems that can automatically handle telephone line activity, such as answering the telephone after a certain number of rings. The **no modem ri-is-cd** command disables a line for a high-speed modem.

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

Example

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

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

notify

To enable terminal notification about pending output from other connections, use the **notify** line configuration command. The 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. The **no notify** command ends notification.

notify
no notify

Syntax Description

These commands have no arguments or keywords.

Default

Disabled

Command Mode

Line configuration

Example

The following example sets up notification of pending output on line 5:

```
line 5
notify
```

padding

To set the padding on a specific output character, use the **padding** line configuration command. The **no padding** command removes padding for the specified output character.

padding *ASCII-number count*
no padding *ASCII-number*

Syntax Description

ASCII-number	The ASCII decimal representation of the character
count	The number of NULL bytes sent after that character, up to 255 padding characters in length

Default

None

Command Mode

Line configuration

Example

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

```
line 4 24
padding 13 25
```

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
mark	Mark

Default

The **none** keyword—no parity

Command Mode

Line configuration

Example

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

```
line 34
parity even
```

password

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

password *password*
no password

Syntax Description

password Case-sensitive 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. 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

None

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

private

To save configurations between sessions, use the **private** line command. This command ensures that the terminal parameter-setting options the user sets remain in effect between terminal sessions. This behavior is desirable for terminals in private offices. Use the **no private** command to restore the default condition.

private
no private

Syntax Description

This command has no arguments or keywords.

Default

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

Command Mode

Line configuration

Example

The following example sets up line 15 to keep all user-supplied settings at system restarts:

```
line 15
private
```

refuse-message

To define a “line-in-use” message, use the **refuse-message** line command. Use the **no refuse-message** command to disable the message.

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

Syntax Description

d	A delimiting character of your choice—a pound sign (#) for example. You cannot use the delimiting character in the message.
message	The text of the “line-in-use” message.

Default

None

Command Mode

Line configuration

Usage Guidelines

Follow the 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 “lines-in-use” message, and the user is given information about what to try next.:

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

Please try again later, or try the slower speed modem named turtle./
```

rotary

To define each group of lines, use the **rotary** line configuration command. This command adds a line to the specified rotary group. To list the defined rotary groups, use the privileged EXEC command **show line**. Use the **no rotary** command to disable the rotary group.

rotary *group*
no rotary

Syntax Description

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

Default

None

Command Mode

Line configuration

Usage Guidelines

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 automatically avoid inactive host ports. To enable this feature, use the **modem cts-required** 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 rotary groups.
- Session timeout—Use the **session-timeout** 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. 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 1-2 shows the services and port numbers for rotary groups and lines.

Table 1-2 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 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 callout
session-timeout

session-limit

To set the maximum number of terminal sessions per line, use the **session-limit** line configuration command. The **no session-limit** command removes 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 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. The **no session-timeout** command removes the timeout definition; however, you can specify a session timeout on each port.

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

Syntax Description

minutes Specifies the time 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.

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):

```
session-timeout 20
```

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

```
session-timeout 10 output
```

Related Command

rotary

special-character-bits

To configure the number of characters used in special characters such as software flow control, escape characters, and so forth, 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 ASCII character set

Default

7-bit ASCII character set

Command Mode

Line configuration

Usage Guidelines

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

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 global configuration command and the line commands.

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

Related Commands

default-value exec-character-bits

default-value special-character-bits

exec-character-bits

terminal data-character-bits

terminal exec-character-bits

terminal special-character-bits

speed

To set the terminal baud rate (speed of the lines), use one of the following **speed** line configuration commands. The **speed** command sets both the transmit (to terminal) and receive (from terminal) speeds. The **txspeed** command sets the transmit speed only. The **rxspeed** command sets the receive speed only.

speed *bps*
txspeed *bps*
rxspeed *bps*

Syntax Description

bps The baud rate in bits per second, see Table 1-3 for settings

Default

9600 bits per second

Command Mode

Line configuration

Usage Guidelines

Use the following table as a guide for setting the line speeds.

Table 1-3 Communication Server Line Speeds in Bits per Second

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

You can set a line to a standard speed with no restrictions. If you set a line to a nonstandard speed, the line that it is paired with must be set to either a standard speed or the same nonstandard speed. The lines are paired as 1/2, 3/4, 5/6, and so forth.

Example

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

```
line 1 2
speed 38400
```

start-character

To set the flow control start character, use the **start-character** line configuration command. The command defines the character that signals the start of data transmission when software flow control is in effect. The **no start-character** command removes the character.

start-character *ASCII-number*
no start-character

Syntax Description

ASCII-number	The ASCII decimal representation of the start character

Default

Ctrl-Q (ASCII character 17)

Command Mode

Line configuration

Example

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

```
line 25
start-character 2
```

state-machine

To specify the transition criteria for the state of a particular state machine, use the **state-machine** global configuration command. With the protocol translator, this command is only useful in two-step translations.

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 only have a single 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... lastcha	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	Defines the state to enter if the character is in the specified range.
transmit	Causes the packet to be transmitted, and the state machine to be reset to state zero. Characters that recur and have not been explicitly defined to have a particular action, return the state machine to state zero.

Default

None

Command Mode

Global configuration

Usage Guidelines

This command is entered with the **dispatch-machine** line 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 all lumped together into a single packet. Because the default is to remain in state 0 without transmitting anything, The example allows normal key signals to be transmitted immediately.

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

Related Command

dispatch-character
dispatch-machine
dispatch-timeout

stop-character

To set the flow control stop character, use the **stop-character** line configuration command. The command defines the character that signals the end of data transmission when software flow control is in effect. The **no stop-character** command removes the character.

stop-character *ASCII-number*
no stop-character

Syntax Description

ASCII-number	The ASCII decimal representation of the stop character

Default

Ctrl-S (ASCII character 19)

Command Mode

Line configuration

Example

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

```
line 25
stop-character 5
```

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 2 stop bits

Default

2 stop bits

Command Mode

Line configuration

Example

The following example changes the default from 2 stopbits to 1 as a performance enhancement.

```
line 33
stopbits 1
```

terminal data-character-bits

To locally set the ASCII character set sent over network connections, use the **terminal data-character-bits** EXEC command.

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

Syntax Description

7 Selects the 7-bit ASCII character set.

8 Selects the full 8-bit ASCII character set.

Default

7 bits

Command Mode

EXEC

Usage Guidelines

Configuring the data character width to 8 bits allows you to add special graphical and international characters in banners, prompts, and so forth. When the user exits the system, the character width is reset to the default value.

Example

The following example temporarily configures a communication server to use a full 8-bit ASCII character set to use the additional graphical and international characters available.

```
CS> terminal data-character-bits 8
```

Related Commands

default-value exec-character-bits

default-value special-character-bits

exec-character-bits

special-character-bits

terminal exec-character-bits

terminal special-character-bits

terminal exec-character-bits

To change the ASCII character widths for characters entered at the EXEC and during configuration mode, use the **terminal exec-character-bits** EXEC command.

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

Syntax Description

7 Selects the 7-bit ASCII character set

8 Selects the full 8-bit ASCII character set

Default

7 bits

Command Mode

EXEC

Usage Guidelines

This EXEC command overrides the **default-value exec-character-bits** global configuration command. Configuring the EXEC character width to 8 bits allows you to add special graphical and international characters in banners, prompts, and so forth.

When the user exits the system, the character width is reset to the default value established by the global configuration command. However, setting the EXEC character width to eight 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 eight bits, although the eighth bit is not needed for the **help** command.

Example

The following example temporarily configures a communication server to use a full 8-bit user interface for system banners and prompts. This allows the use of additional graphical and international characters.

```
CS> terminal exec-character-bits 8
```

Related Commands

default-value exec-character-bits

default-value special-character-bits

exec-character-bits

special-character-bits

terminal data-character-bits

terminal special-character-bits

terminal monitor

To copy **debug** output to the current terminal line, use the terminal monitor command.

terminal monitor

Syntax Description

This command has no arguments or keywords.

Default

Disabled.

Command Mode

Privileged EXEC

Usage Guidelines

By default, error messages are directed to the system console, but they can be directed to other devices.

Example

In the following example, error messages will be copied to, and displayed on, a nonconsole terminal:

```
cs# terminal monitor
```

Related Commands

logging buffered (†)
logging host(†)
no logging on(†)

terminal special-character-bits

To temporarily change the ASCII character widths to accept special characters entered during data connection, use the **terminal special-character-bits** EXEC command.

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

Syntax Description

7 Selects the 7-bit ASCII character set.

8 Selects the full 8-bit ASCII character set.

Default

7 bits

Command Mode

EXEC

Usage Guidelines

The **terminal special-character-bits** command temporarily allows the server to support international character sets. It overrides the **default-value special-character-bits** global configuration command and is used to compare character sets typed by the user with the special character available during a data connection, which includes software flow control and escape characters. Configuring the width to 8 allows you to use twice as many special characters as with the 7-bit setting. When the user exits the system, the command is reset to the default value established by the global configuration command.

Example

The following example temporarily configures full 8-bit comparisons of flow control and interrupt characters to allow more special characters to be accepted. When you exit the system, character width will be reset to the width established by the global configuration command.

```
CS> terminal special-character-bits 8
```

Related Commands

default-value exec-character-bits

default-value special-character-bits

exec-character-bits

special-character-bits

terminal data-character-bits

terminal exec-character-bits

terminal-type

To specify the type of terminal connected to the line, use the **terminal-type** line configuration command. The command records the type of terminal connected to the line. The **no terminal-type** command removes any information about the type of terminal and resets the line to the default terminal emulation.

terminal-type *terminal-name*
no terminal-type

Syntax Description

terminal-name	Terminal name and type. For the TN3270 application, this argument specifies a defined termcap name.

Default

VT100

Command Mode

Line configuration

Usage Guidelines

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 type VT220:

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

transport input

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

transport input [**telnet** | **lat** | **pad** | **none**]

Syntax Description

telnet	Specifies all types of incoming TCP/IP connections.
lat	Selects the Digital LAT protocol, and specifies both incoming reverse LAT and host-initiated connections.
pad	Selects X.3 PAD incoming connections.
none	Prevents any protocol selection on the line. This makes the port unusable by incoming connections.

Default

All supported protocols allowed on the line

Command Mode

Line configuration

Usage Guidelines

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 also useful in controlling exactly which protocols can be translated to other protocols when using two-step translation.

Access lists for each individual protocol may be defined in addition to the allowances created by the **transport input** command.

Example

The following example sets the preferred incoming protocol to LAT:

```
line vty 0 32
transport input lat
```

The following example sets specific lines for specific protocol translation:

```
! For protocol translation, reserve approximately 1/3 of the available
! lines for each possible transport protocol. Also prevent users from
! "hopping" off with the same protocol they came in with.
line vty 0 32
transport input lat
transport output telnet pad
!
line vty 33 65
transport input telnet
transport output pad lat
!
line vty 66 99
transport input pad
transport output lat telnet
```

Related Commands

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 [**telnet | lat | rlogin | pad | none**]

Syntax Description

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.
lat	Selects the Digital LAT protocol, the Local Area Transport protocol used most often to connect communication servers to DEC 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.
pad	Selects X.3 PAD, used most often to connect communication servers to X.25 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.

Default

All supported protocols allowed on the line

Command Mode

Line configuration

Example

The following example illustrates how to protect inbound connections:

```
!
! On a communication server, don’t allow any inbound connections to the
! ports that are actual terminals. This prevents trojan horse programs
! from attaching to the port and deriving passwords.
!
line 1 20
location Undergrad terminal room
transport input none
!
```

Related Commands

transport input
transport preferred

transport preferred

To specify the preferred protocol to use when a command does not specify one, use the **transport preferred** line configuration command.

transport preferred [**telnet** | **lat** | **rlogin** | **pad** | **none**]

Syntax Description

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.

lat Selects the Digital LAT protocol, the Local Area Transport protocol used most often to connect communication servers to DEC 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.

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

Default

For communication servers that support LAT, the default protocol is LAT. For those that do not support LAT, the default is Telnet.

Command Mode

Line configuration

Example

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

```
!
line tty 1
transport preferred rlogin
!
```

Related Commands

transport input
transport output

vacant-message

To display an idle terminal message, use the **vacant-message** line configuration command. The 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. The **no vacant-message** command removes 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	A delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
message	The 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

Follow the 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 the 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 columns

Command Mode

Line configuration

Usage Guidelines

The rlogin protocol uses the *characters* argument to set up terminal parameters on a remote UNIX 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 columns to 132 for the terminal on line 7:

```
line 7
location terminal3
width 132
```