



Managing Connections, Menus, and System Banners

This chapter describes how to manage connections to other hosts, set banner messages for router users, and create menus of specific user tasks. For a complete description of the connections, menu, and system banner commands in this chapter, refer to the “Connection, Menu, and System Banner Commands” chapter in the *Cisco IOS Configuration Fundamentals Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

The following sections describe the tasks required to manage connections, configure messages and banners, and configure user-menus:

- Managing Connections
- Setting Up Terminal Messages
- Setting Up Terminal Banners and Banner Tokens
- Creating Menus

Examples for the above sections can be found at the end of the chapter in the “Connection Management, System Banner, and User-Menu Examples” section.

Managing Connections

This section describes session-management activities. The following sections describe connection-management activities that apply to all supported connection protocols:

- Escaping to the EXEC Prompt
- Switching to Another Connection
- Assigning a Logical Name to a Connection
- Changing a Login Name
- Locking Access to a Terminal
- Specifying a TACACS Host
- Sending Messages to Other Terminals
- Clearing TCP/IP Connections
- Exiting a Session Started from a Router

- Logging Out of a Router
- Disconnecting a Line

Escaping to the EXEC Prompt

After you have started a connection, you can escape out of the current session and return to the EXEC prompt by using the escape sequence command (**Ctrl-Shift-6** then **x** [**Ctrl^x**] by default). You can type the command character as you hold down the **Ctrl** key or with the **Ctrl** key released; you can type either uppercase or lowercase letters.



Note

In screen output examples that show two caret (^) symbols together, the first caret represents the Control key (**Ctrl**) and the second caret represents the keystroke sequence **Shift-6**. The double-caret combination (^) means hold down the **Ctrl** key while you press the **Shift** and the **6** key.

By default, the escape sequence is **Ctrl^x**. If you press the escape key (**Escape-Char**), you change the **Shift-Ctrl-6** sequence to whatever you want. For example, if you press **Escape-Char Break**, the **Break** key becomes the new escape character to suspend a session and to access the EXEC prompt.

Switching to Another Connection

You can have several concurrent sessions open and switch back and forth between them.

The number of sessions that can be open is defined by the **session-limit** command.

To switch between sessions by escaping one session and resuming a previously opened session, use the following commands:

	Command	Purpose
Step 1	Ctrl-Shift-6 then x (Ctrl^x) by default	Escapes the current connection and return to the EXEC prompt.
Step 2	where	Lists the open sessions. All open sessions associated with the current terminal line are displayed.
Step 3	resume [<i>connection</i>] [<i>keyword</i>]	Makes the connection.

The **Ctrl^x**, **where**, and **resume** commands are available with all supported connection protocols.

You could also make a new connection while you are at the EXEC prompt.

Assigning a Logical Name to a Connection

To assign a logical name to a connection, use the following command in EXEC mode:

Command	Purpose
name-connection	Assigns a logical name to a connection.

The logical name can be useful for keeping track of multiple connections.

You are prompted for the connection number and name to assign. The **where** command displays a list of the assigned logical connection names.

Changing a Login Name

You can change a login username if you must match outgoing access list requirements or other login prompt requirements. To change a login username, use the following command in user EXEC mode:

Command	Purpose
<code>login</code>	Changes a login username.

When you enter this command, the system prompts you for a username and password. Enter the new username and the original password. If the username does not match, but the password does, the Cisco IOS software updates the session with the new username used by **login** command attempt.

If no username and password prompts appear, the network administrator did not specify that a username and password be required at login time. If both the username and password are entered correctly, the session becomes associated with the specified username.

When you access a system with TACACS security, you can enter your login name or specify a TACACS server by using the following argument when the “Username:” prompt appears:

```
user @tacacs-server
```

The router must be one of the routers defined in a router configuration. For more information, refer to the “Specifying a TACACS Host” section later in this chapter, or refer to the **tacacs-server host** command in the “TACACS, Extended TACACS, and TACACS+ Commands” chapter of the *Security Command Reference*.

If you do not specify a host, the router tries each of the TACACS servers in the list until it receives a response.

If you specify a host that does not respond, no other TACACS server will be queried. The router either denies access or function, according to the action specified by the **tacacs-server last-resort** command, if it is configured.

If you specified a TACACS server host with the *user @tacacs-server* argument, the TACACS server specified is used for all subsequent authentication or notification queries, with the possible exception of SLIP address queries.

For an example of changing a login name, see the “Changing a Login Name Example” section at the end of this chapter.

Locking Access to a Terminal

You can prevent access to your terminal session while keeping your connection open by setting up a temporary password. To lock access to the terminal, use the following commands in EXEC mode:

	Command	Purpose
Step 1	<code>lock</code>	Issues the lock command. The system prompts you for a password.
Step 2	<code>password</code>	Enters a password, which can be any arbitrary string. The screen clears and displays the message “Locked.”
Step 3	<code>password</code>	To regain access to your sessions, re-enters the password.

The Cisco IOS software honors session timeouts on a locked line. You must clear the line to remove this feature. The system administrator must set up the line to allow use of the temporary locking feature.

Specifying a TACACS Host

You can specify a TACACS host when you dial in or use the **login** command. Only the specified host is accessed for user authentication information.

To specify the name of a TACACS host at login, use the following command in EXEC mode:

Command	Purpose
<code>user@hostname</code>	Specifies the name of a TACACS host at login.

For an example of specifying a TACACS host, see the “Specifying a TACACS Host Example” section at the end of this chapter.

Sending Messages to Other Terminals

You can send messages to one or all terminal lines. A common reason for doing this is to inform users of an impending shutdown. To send a message to other terminals, use the following command in EXEC mode:

Command	Purpose
<code>send {line-number *}</code>	Sends a message to other terminals.

The system prompts for the message, which can be up to 500 characters long. Enter **Ctrl-Z** to end the message. Enter **Ctrl-C** to abort the command.

Clearing TCP/IP Connections

To clear a TCP connection, use the following command in privileged EXEC mode:

Command	Purpose
<code>clear tcp {line <i>line-number</i> local <i>host-name port remote host-name port</i> tcb <i>address</i>}</code>	Clears a TCP connection.

The **clear tcp** command is particularly useful for clearing hung TCP connections.

The **clear tcp line *line-number*** command terminates the TCP connection on the specified TTY line. Additionally, all TCP sessions initiated from that TTY line are terminated.

The **clear tcp local *host-name port remote host-name port*** command terminates the specific TCP connection identified by the host name/port pair of the local and remote router.

Exiting a Session Started from a Router

The protocol used to initiate a session determines how you exit that session.

To exit XRemote, you must quit all active X connections, usually with a command supported by your X client system. Usually, when you quit the last connection (all client processes are stopped), XRemote closes and you return to the EXEC prompt. Check your X client system documentation for specific information about exiting an XRemote session.

To exit a SLIP and PPP, you must hang up the dial-in connection, usually with a command that your dial-in software supports.

To exit a LAT, Telnet, rlogin, TN3270, and X.3 PAD session begun from the router to a remote device, enter the escape sequence (**Ctrl-Shift-6** then **x** [**Ctrl^x**] by default) and enter the **disconnect** command at the EXEC prompt. You can also log off the remote system.

Except for XRemote, you also can escape to the EXEC prompt and enter either of the following commands to terminate an active terminal session:

- **exit**
- **logout**

To exit a Telnet session *to* a router, see the “Log Out of a Router” section.

Logging Out of a Router

The method you use to disconnect from a router depends on where you are located in relation to the router, and the port on the router to which you log in. Keep the following in mind:

- If your terminal or computer running a terminal-emulation application is connected physically to the console port of the router, you can disconnect from the console port by physically disconnecting the cable from the console port of the router.
- If your terminal or computer running a terminal-emulation application is remotely connected to the console port of the router, you disconnect by issuing the command or key sequence used by your terminal-emulation package. For example, if you are on a Macintosh computer running the application “TCP/Connect” from InterCon Corporation, you would press **Ctrl-]** at the user or privileged EXEC prompt to disconnect.

- If you are on a remote terminal and connect to a VTY line through a synchronous interface on the router, you can issue one of the following commands to disconnect:
 - **exit**
 - **logout**

Disconnecting a Line

To disconnect a line, use the following command in EXEC mode:

Command	Purpose
<code>disconnect</code> [<i>connection</i>]	Disconnects a line.

Avoid disconnecting a line to end a session. Instead, log off the host to allow the router to clear the connection. Then end the session. Only if you cannot log out of an active session should you disconnect the line.

Setting Up Terminal Messages

The types of messages that can be displayed to terminal users who connect to the router are described in the following sections:

- Configuring an Idle Terminal Message
- Displaying a “Line in Use” Message
- Displaying a “Host Failed” Message

Configuring an Idle Terminal Message

You can configure messages to be displayed on a console or terminal not in use. Also called a *vacant message*, this message is different from the banner message displayed when an EXEC process is activated. To configure an idle terminal message, use the following command in line configuration mode:

Command	Purpose
<code>vacant-message</code> [<i>d message d</i>]	Displays an idle terminal message.

Displaying a “Line in Use” Message

You can display a “line in use” message when an incoming connection is attempted and all rotary group or other lines are in use. Use the following command in line configuration mode:

Command	Purpose
<code>refuse-message</code> <i>d message d</i>	Displays a “line in use” message.

If you do not define such a message, the user receives a system-generated error message when all lines are in use. You also can use this message to provide the user with further instructions.

Displaying a “Host Failed” Message

You can display a “host failed” message when a Telnet connection with a specific host fails. Use the following command in line configuration mode:

Command	Purpose
<code>busy-message hostname d message d</code>	Displays a “host failed” message.

Setting Up Terminal Banners and Banner Tokens

Banners can be customized with the use of banner tokens as described in the following section:

- Using Banner Tokens

The types of banners that can be displayed to terminal users who connect to the router are described in the following sections:

- Configuring a Message-of-the-Day Banner
- Configuring a Login Banner
- Configuring a Line-Activation (EXEC) Banner
- Configuring an Incoming Banner
- Configuring an SLIP-PPP Banner Message

You also can turn off message displays, as described in the “Enabling or Disabling the Display of Banners” section.

For an example of displaying terminal banner messages, see the “Configuring Banners Example” section at the end of this chapter.

Using Banner Tokens

Banner tokens enable you to use tokens with all configurable Cisco IOS banner commands. Tokens are keywords of the form `$(token)` that, when used in a banner message, display the currently configured value of the token argument (for example the router’s hostname, domain name, or IP address). Using these tokens, you can design your own banners that will display current IOS configuration variables. Only Cisco IOS supported tokens may be used. There is no facility for you to define your own tokens.

Table 5 lists the tokens supported by the different **banner** commands.

Table 5 Tokens Allowed By Banner-Type

Token		motd banner	login banner	exec banner	incoming banner	slip-ppp banner
<code>\$(hostname)</code>	Router’s Host Name	YES	YES	YES	YES	YES
<code>\$(domain)</code>	Router’s Domain Name	YES	YES	YES	YES	YES

Table 5 Tokens Allowed By Banner-Type (continued)

Token		motd banner	login banner	exec banner	incoming banner	slip-ppp banner
\$(peer-ip)	IP Address of the Peer Machine	NO	NO	NO	NO	YES
\$(gate-ip)	IP Address of the Gateway Machine	NO	NO	NO	NO	YES
\$(encap)	Encapsulation Type (SLIP or PPP)	NO	NO	NO	NO	YES
\$(encap-alt)	Encapsulation Type Displayed as SL/IP instead of SLIP	NO	NO	NO	NO	YES
\$(mtu)	Maximum Transmission Unit Size	NO	NO	NO	NO	YES
\$(line)	VTY or TTY (async) Line Number	YES	YES	YES	YES	NO
\$(line-desc)	Description Attached to the Line	YES	YES	YES	YES	NO

Configuring a Message-of-the-Day Banner

You can configure a message-of-the-day (MOTD) banner to be displayed on all connected terminals. This banner is displayed at login and is useful for sending messages that affect all network users (such as impending system shutdowns). To do so, use the following command in global configuration mode:

Command	Purpose
<code>banner motd d message d</code>	Configures a MOTD banner.

Configuring a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner is displayed after the MOTD banner and before the login prompts.

To configure a login banner, use the following command in global configuration mode:

Command	Purpose
<code>banner login d message d</code>	Configures a login banner.

The login banner cannot be disabled on a per-line basis. To globally disable the login banner, you must delete the login banner with the **no banner login** command.

Configuring a Line-Activation (EXEC) Banner

You can configure a line-activation banner to be displayed when an EXEC process (such as a line-activation or incoming connection to a VTY line) is created. To do so, use the following command in global configuration mode:

Command	Purpose
<code>banner exec d message d</code>	Configures a banner to be displayed on terminals with an interactive EXEC session.

Configuring an Incoming Banner

You can configure a banner to be displayed on terminals connected to reverse Telnet lines. This banner is useful for providing instructions to users of these types of connections. Reverse Telnet connections are described in more detail in the “Establishing a Reverse Telnet Session to a Modem” chapter in the *Dial Solutions Configuration Guide*.

To configure a banner that is sent on incoming connections, use the following command in global configuration mode:

Command	Purpose
<code>banner incoming d message d</code>	Configures a banner to display on terminals connected to reverse Telnet lines.

Configuring an SLIP-PPP Banner Message

Default banner messages have been known to cause connectivity problems in some non-Cisco Serial Link Interface Protocol (SLIP) and PPP dial-up software. You can now customize the SLIP-PPP banner message to make Cisco SLIP and PPP compatible with non-Cisco dial-up software. To configure the SLIP-PPP banner message, use the following command in global configuration mode:

Command	Purpose
<code>banner slip-ppp d message d</code>	Configures the SLIP-PPP banner to display a customized message.

Enabling or Disabling the Display of Banners

You can control display of the MOTD and line-activation (EXEC) banners. By default, these banners are displayed on all lines. To suppress or reinstate the display of such banners, use one of the following commands in line configuration mode:

Command	Purpose
<code>no exec-banner</code>	Suppresses MOTD and EXEC banner display.
<code>exec-banner</code>	Reinstates the display of the EXEC or MOTD banners.
<code>no motd-banner</code>	Suppresses MOTD banner display only.
<code>motd-banner</code>	Reinstates the display of the MOTD banners.

These commands determine whether the router will display the EXEC banner and the MOTD banner when an EXEC session is created. These banners are defined with the **banner motd** and **banner exec** commands. By default, the MOTD banner and the EXEC banner are enabled on all lines.

Disable the EXEC and MOTD banners using the **no exec-banner** command.

The MOTD banners can also be disabled by the **no motd-banner** line configuration command, which disables MOTD banners on a line. If the **no exec-banner** command is configured on a line, the MOTD banner will be disabled regardless of whether the **motd-banner** command is enabled or disabled. Table 6 summarizes the effects of the **exec-banner** command and the **motd-banner** command.

Table 6 *Banners Displayed*

	exec-banner (default)	no exec-banner
motd-banner (default)	MOTD banner	None
	EXEC banner	
no motd-banner	EXEC banner	None

For reverse Telnet connections, the EXEC banner is never displayed. Instead, the incoming banner is displayed. The MOTD banner is displayed by default, but it is disabled if either the **no exec-banner** command or **no motd-banner** command is configured. Table 7 summarizes the effects of the **exec-banner** command and the **motd-banner** command for reverse Telnet connections.

Table 7 *Banners Displayed—Reverse Telnet Session to Async Lines*

	exec-banner (default)	no exec-banner
motd-banner (default)	MOTD banner	incoming banner
	incoming banner	
no motd-banner	incoming banner	incoming banner

Creating Menus

A menu is a displayed list of actions from which a user can select without having to know anything about the underlying command-level details. A menu system (also known as a user-menu) effectively controls which functions a user can access. Figure 5 illustrates the parts that make up a typical menu.

Figure 5 *Typical Menu Example*



Creating a Menu Task List

To create menus, perform the tasks in the following sections:

- Understanding Menu Guidelines
- Specifying the Menu Title
- Specifying the Menu Prompt
- Specifying the Menu Item Text
- Specifying the Underlying Command for the Menu Item
- Specifying the Default Command for the Menu
- Creating a Submenu
- Creating Hidden Menu Entries
- Specifying Menu Display Configuration Options
- Specifying per-Item Menu Options
- Invoking the Menu
- Deleting the Menu from the Configuration

Understanding Menu Guidelines

Anyone who can enter configuration mode can create these menus. Keep the following guidelines in mind when you create menus:

- Each menu item represents a single user command.
- The menu system default is a standard “dumb” terminal that only displays text in a 24-line-by-80-column format.

- A menu can have a maximum of 18 menu items. Menus containing more than 9 menu items are automatically configured as single-spaced menus; menus containing 9 or fewer menu items are automatically configured as double-spaced menus, but can be configured as single-spaced menus using the **menu single-space** command. (For more information about menu display configuration options, refer to the section “Specifying Menu Display Configuration Options” later in this chapter.)
- Item keys can be numbers, letters, or strings. If you use strings, you must configure the **menu line-mode** command.
- When you construct a menu, always specify how a user exits a menu and where the user goes. If you do not provide an exit from a menu—such as with the **menu-exit** command (described in the section “Specifying the Underlying Command for the Menu Item” later in this section), there is no way to exit the menu.
- The **exec-timeout** command can be used to close and clean up an idle menu; the **session-timeout** command can be used to clean up a menu with an open connection.

Specifying the Menu Title

You can specify an identifying title for the menu. To specify the menu title, use the following command in global configuration mode:

Command	Purpose
<code>menu name title d title d</code>	Specifies the title for the menu. The <i>d</i> indicates a delimiting character.

The following example specifies the title that is displayed when the OnRamp menu is selected. The following four main elements create the title:

- The **menu title** command
- Delimiter characters that open and close the title text
- Escape characters to clear the screen (optional)
- Title text

The following example shows the command used to create the title for the menu shown in Figure 3, at the beginning of this section:

```
Router(config)# menu OnRamp title /^[H^[[J
Enter TEXT message. End with the character '/'.
    Welcome to OnRamp Internet Services

    Type a number to select an option;
    Type 9 to exit the menu.
/
Router(config)#
```

You can position the title of the menu horizontally by preceding the title text with blank characters. You can also add lines of space above and below the title by pressing Enter.

In this example, the title text consists of the following:

- One-line title
- Space
- Two-line menu instruction banner

Title text must be enclosed within text delimiter characters—the slash character (/) in this example. Title text delimiters are characters that do not ordinarily appear within the text of a title, such as slash (/), double quote ("), or tilde (~). You can use any character that is not likely to be used within the text of the title as delimiter characters. Ctrl-C is reserved for special use and should not be used in the text of the title.

This title text example also includes an escape character sequence to clear the screen before displaying the menu. In this case the string `^[H^[J` is an escape string used by many VT100-compatible terminals to clear the screen. To enter it, you must enter **Ctrl-V** before each escape character (^).

You can also use the **menu clear-screen** command to clear the screen before displaying menus and submenus, instead of embedding a terminal-specific string in the menu title. This option uses a terminal-independent mechanism based on termcap entries defined in the router and the terminal type configured for the user's terminal. The **menu clear-screen** command allows the same menu to be used on multiple types of terminals instead of having terminal-specific strings embedded within menu titles. If the termcap entry does not contain a clear string, the menu system inserts 24 new lines, causing all existing text to scroll off the top of the terminal screen.

To clear the screen before displaying the menu, use the following command in global configuration mode:

Command	Purpose
<code>menu name clear-screen</code>	Specifies screen clearing before displaying menus and submenus.

The following example clears the screen before displacing the OnRamp menu or a submenu:

```
Router(config)# menu OnRamp clear-screen
```

Specifying the Menu Prompt

You can specify a prompt for the menu. To specify the menu prompt, use the following command in global configuration mode:

Command	Purpose
<code>menu name prompt delimiter prompt delimiter</code>	Specifies the prompt for the menu.

Specifying the Menu Item Text

Each displayed menu entry consists of the selection key (number, letter, or string) and the text describing the action to be performed. You can specify descriptive text for a maximum of 18 menu items. Because each menu entry represents a single user interface command, you must specify the menu item text one entry at a time. To specify the menu item text, use the following command in global configuration mode:

Command	Purpose
<code>menu name text item text</code>	Specifies the text for the menu item.

The following example specifies the text that is displayed for the three entries in the OnRamp menu:

```
Router(config)# menu OnRamp text 1 Read email
Router(config)# menu OnRamp text 2 UNIX Internet Access
Router(config)# menu OnRamp text 9 Exit menu system
```

You can provide access to context-sensitive help by creating a “help server” host and use a menu entry to make a connection to that host.

Menu selection keys do not need to be contiguous. You can provide consistency across menus by assigning a particular number, letter, or string to a special function—such as Help or Exit—regardless of the number of menu entries in a given menu. For example, menu entry H could be reserved for help across all menus.

When more than nine menu items are defined in a menu, the **menu line-mode** and **menu single-space** commands are activated automatically. The commands can be configured explicitly for menus of nine items or fewer. For more information on these commands, refer to the section “Specifying Menu Display Configuration Options” later in this chapter.

Specifying the Underlying Command for the Menu Item

Each displayed menu entry issues a user interface command when the user enters its key. Each menu entry can have only a single command associated with it. To specify the menu item command, use the following command in global configuration mode:

Command	Purpose
<code>menu name command item command</code>	Specifies the command to be performed when the menu item is selected.

The following example specifies the commands that are associated with the three entries in the OnRamp menu:

```
Router(config)# menu OnRamp command 1 rlogin mailsys
Router(config)# menu OnRamp command 2 rlogin unix.cisco.com
Router(config)# menu OnRamp command 9 menu-exit
```

The **menu-exit** command is available only from within menus. This command provides a way to return to a higher-level menu or to exit the menu system.

When a menu item allows you to make a connection, the menu item should also contain entries that can be used to resume connections; otherwise, when you try to escape from a connection and return to the menu, there is no way to resume the session. It will sit idle until you log off.

You can build the **resume connection** EXEC command into a menu entry so that the user can resume a connection, or you can configure the line using the **escape-char none** command to prevent users from escaping their sessions.

To specify connection resumption as part of the menu item command, use the following command in global configuration mode:

Command	Purpose
<code>menu name command item resume [connection] /connect [connect string]</code>	Specifies the command to be performed when the menu item is selected.

Embedding the **resume** command within the **menu** command permits a user to resume the named connection or make another connection using the specified name, if there is no active connection by that name. As an option, you can also supply the connect string needed to connect initially. When you do not supply this connect string, the command uses the specified connection name.

You can use the **resume** command in the following menu entries:

- Embedded in a menu entry
- As a separate, specific menu entry
- As a “rotary” menu entry that steps through several connections

In the following example, the **resume** command is embedded in the **menu** command so that selecting the menu item either starts the specified connection session (if one is not already open) or resumes the session (if one is already open):

```
Router(config)# menu Duluth text 1 Read email
Router(config)# menu Duluth command 1 resume mailsys /connect rlogin mailsys
```

In the following example, the **resume** command is used in a separate menu entry (entry 3) to resume a specific connection:

```
Router(config)# menu Duluth text 3 Resume UNIX Internet Access
Router(config)# menu Duluth command 3 resume unix.cisco.com
```

You use the **resume/next** command to resume the next open connection in the user’s list of connections. This command allows you to create a single menu entry that steps through all of the user’s connections. To specify **resume/next** connection resumption as part of the menu item command, use the following command in global configuration mode:

Command	Purpose
<code>menu name command item resume /next</code>	Specifies resume/next connection resumption.

The following example shows a menu entry (entry 6) created to step through all of the user’s connections:

```
Router(config)# menu Duluth text 6 Resume next connection
Router(config)# menu Duluth command 6 resume /next
```

Specifying the Default Command for the Menu

When a user presses Enter without specifying an item, the router performs the command for the default item. To specify the default item, use the following command in global configuration mode:

Command	Purpose
<code>menu name default item</code>	Uses the command to be performed when no item is specified.

Creating a Submenu

To create submenus that are opened by selecting a higher-level menu entry, use the **menu** command to invoke a menu in a line menu entry. To specify a submenu item command, use the following commands in global configuration mode:

	Command	Purpose
Step 1	<code>menu name text item text</code>	Specifies the menu item that invokes the submenu.
Step 2	<code>menu name command item menu name2</code>	Specifies the command to be used when the menu item is selected.
Step 3	<code>menu name2 title delimiter title2 delimiter</code>	Specifies the title for the submenu.
Step 4	<code>menu name2 text item text</code>	Specifies the submenu item.
Step 5	<code>menu name2 command item command</code>	Specifies the commands to be used when the submenu item is selected.

The following example specifies that the menu item (entry 8) activates the submenu in the OnRamp menu:

```
Router(config)# menu OnRamp text 8 Set terminal type
```

The following example specifies the command that is performed when the menu item (entry 8) is selected in the OnRamp menu:

```
Router(config)# menu OnRamp command 8 menu Terminals
```

The following example specifies the title for the Terminals submenu:

```
Router(config)# menu Terminals title /
                Supported Terminal Types
```

```
                Type a number to select an option;
                Type 9 to return to the previous menu.
```

The following example specifies the submenu items for the Terminals submenu:

```
Router(config)# menu Terminals text 1 DEC VT420 or similar
Router(config)# menu Terminals text 2 Heath H-19
Router(config)# menu Terminals text 3 IBM 3051 or equivalent
Router(config)# menu Terminals text 4 Macintosh with gterm emulator
Router(config)# menu Terminals text 9 Return to previous menu
```

The following example specifies the commands associated with the items in the Terminals submenu:

```
Router(config)# menu Terminals command 1 term terminal-type vt420
Router(config)# menu Terminals command 2 term terminal-type h19
Router(config)# menu Terminals command 3 term terminal-type ibm3051
Router(config)# menu Terminals command 4 term terminal-type gterm
Router(config)# menu Terminals command 9 menu-exit
```

When you select entry 8 on the main menu, the Terminals submenu appears:

```

Supported Terminal Types

Type a number to select an option;
Type 9 to return to the previous menu.

1    DEC VT420 or similar
2    Heath H-19
3    IBM 3051 or equivalent
4    Macintosh with gterm emulator
9    Return to previous menu

```



Note

If you nest too many levels of menus, the system prints an error message on the terminal and returns to the previous menu level.

Creating Hidden Menu Entries

A hidden menu entry is a menu item that contains a selection key but no associated text describing the action to be performed. Include this type of menu entry to aid system administrators who help users. The normal procedure is to specify a menu command but omit specifying any text for the item. To specify a hidden menu item, use the following command in global configuration mode:

Command	Purpose
<code>menu name command item command</code>	Specifies the command to be used when the hidden menu entry is selected.

The following example shows the command associated with the submenu entry in the OnRamp menu:

```
Router(config)# menu OnRamp command 7 show whoami
```

The **show whoami** command can be included in menus to aid system administrators who help users. If text is included as an argument in the command, that text is displayed as part of the additional data about the line, and helps identify exactly which menu or submenu the user is accessing. Because the **show whoami** command is hidden inside the menu entry, this information might not be otherwise available. For example, the hidden menu entry created by the line in the configuration file `menu OnRamp command 7 show whoami` Terminals submenu of OnRamp Internet Access menu might display information similar to the following:

```
Comm Server "cs101", Line 0 at 0 bps. Location "Second floor, West"
Additional data: Terminals submenu of OnRamp Internet Access menu
```

To prevent the information from being lost if the menu display clears the screen, this command always displays a More prompt before returning.

Specifying Menu Display Configuration Options

In addition to the **menu clear-screen** command, described in the section “Specifying the Menu Title,” the following are the three other **menu** commands that define menu functions:

- **menu line-mode**
- **menu single-space**
- **menu status-line**

Using Line Mode in Menus

In a menu of nine or fewer items, you ordinarily select a menu item by entering the item number or a letter. In line mode, you select a menu entry by entering the item key and pressing Enter. The line mode allows you to backspace over the selection and enter another before pressing Enter to issue the command. This function allows you to change the selection before you invoke the command.

To invoke the **line-mode** option, use the following command in global configuration mode:

Command	Purpose
menu <i>name</i> line-mode	Specifies line-mode operation.

The line-mode option is invoked automatically when more than nine menu items are defined, but it can also be configured explicitly for menus of nine items or fewer.

In order to use strings as selection keys, you must enable the **menu line-mode** command.

Displaying Single-Spaced Menus

If there are nine or fewer menu items, the Cisco IOS software ordinarily displays the menu items double-spaced. In a menu of more than nine items, the **single-space** option is activated automatically to fit the menu into a normal 24-line terminal screen. However, the single-space option also can be configured explicitly for menus of nine or fewer items.

To invoke the **single-space** option, use the following command in global configuration mode:

Command	Purpose
menu <i>name</i> single-space	Specifies single-space operation.

Displaying an Informational Status Line

The **status-line** option displays a line of status information about the current user at the top of the terminal screen before the menu title is displayed. This status line includes the router’s host name, the user’s line number, and the current terminal type and keymap type (if any).

To display the **status-line** option, use the following command in global configuration mode:

Command	Purpose
menu <i>name</i> status-line	Displays a status line when using a menu.

Specifying per-Item Menu Options

To configure per-item options, use either or both of the following commands in global configuration mode:

Command	Purpose
<code>menu name options item pause</code>	After the command is issued, pauses before redrawing the menu. Enter this command once for each menu item that pauses.
<code>menu name options item login</code>	Requires a login before the command. Enter this command once for each menu item that requires a login.

Invoking the Menu

To invoke the menu, use the following command at the EXEC prompt:

Command	Purpose
<code>menu name</code>	Invokes the menu by specifying the name of the menu.

You can define menus containing privileged EXEC commands, but users must have privileged access when they start up the menu.

To ensure that a menu is automatically invoked on a line, make sure the menu does not have any exit paths that leave users in an interface they cannot operate, then configure that line with the command **autocommand menu** *menu_name*.

Menus also can be invoked on a per-user basis by defining an **autocommand** for that local username.

Invoking a Menu Example

The following example invokes the *OnRamp* menu:

```
Router> menu OnRamp

Welcome to OnRamp Internet Services

Type a number to select an option;
Type 9 to exit the menu.

1 Read email
2 UNIX Internet access
3 Resume UNIX connection

6 Resume next connection

9 Exit menu system
```

Deleting the Menu from the Configuration

To delete the menu from the configuration, use the following command in global configuration mode:

Command	Purpose
<code>no menu name</code>	Deletes the menu by specifying the menu name.

In order to use the menu again, you must reconfigure the entire menu again.

The following example deletes the OnRamp menu from the configuration:

```
Router(config)# no menu OnRamp
```

Connection Management, System Banner, and User-Menu Examples

This section contains the following examples:

- Changing a Login Name Example
- Specifying a TACACS Host Example
- Clearing a TCP/IP Connection Example
- Configuring Banners Example
- Setting a SLIP-PPP Banner with Banner Tokens Example
- Configuring a Menu Example

Changing a Login Name Example

The following example shows how login usernames and passwords can be changed. In this example, a user currently logged in under the username *user1* attempts to change that login name to *user2*. After entering the **login** command, the user enters the new username, but enters an incorrect password. Because the password does not match the original password, the system rejects the attempt to change the username.

```
Router> login
Username: user2
Password:
% Access denied
Still logged in as "user1"
```

Next, the user attempts the login change again, with the username *user2*, but enters the correct (original) password. This time the password matches the current login information, the login username is changed to *user2*, and the user is allowed access to the EXEC at the user-level.

```
router> login
Username: user2
Password:
router>
```

Specifying a TACACS Host Example

In the following example, *user1* specifies the TACACS host *host1* to authenticate the password:

```
router> login
Username: user1@host1
Translating "HOST1"...domain server (131.108.1.111) [OK]
```

Clearing a TCP/IP Connection Example

The following example clears a TCP connection using its TTY line number. The **show tcp** command displays the line number (tty2) that is used in the **clear tcp** command.

```
Router# show tcp

tty2, virtual tty from host router20.cisco.com
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Local host: 171.69.233.7, Local port: 23
Foreign host: 171.69.61.75, Foreign port: 1058

Enqueued packets for retransmit: 0, input: 0, saved: 0

Event Timers (current time is 0x36144):
Timer           Starts    Wakeups          Next
Retrans         4         0                0x0
TimeWait        0         0                0x0
AckHold         7         4                0x0
SendWnd         0         0                0x0
KeepAlive       0         0                0x0
GiveUp          0         0                0x0
PmtuAger        0         0                0x0

iss: 4151109680 snduna: 4151109752 sndnxt: 4151109752  sndwnd: 24576
irs: 1249472001 rcvnxt: 1249472032 rcvwnd: 4258  delrcvwnd: 30

SRTT: 710 ms, RTTO: 4442 ms, RTV: 1511 ms, KRTT: 0 ms
minRTT: 0 ms, maxRTT: 300 ms, ACK hold: 300 ms

Router# clear tcp line 2
[confirm]
[OK]
```

The following example clears a TCP connection by specifying its local router host name and port and its remote router host name and port. The **show tcp brief** command displays the local (Local Address) and remote (Foreign Address) host names and ports to use in the **clear tcp** command.

```
Router# show tcp brief
TCB           Local Address           Foreign Address         (state)
60A34E9C     router1.cisco.com.23    router20.cisco.1055    ESTAB

Router# clear tcp local router1 23 remote router20 1055
[confirm]
[OK]
```

The following example clears a TCP connection using its TCB address. The **show tcp brief** command displays the TCB address to use in the **clear tcp** command.

```
Router# show tcp brief
TCB          Local Address      Foreign Address    (state)
60B75E48    router1.cisco.com.23  router20.cisco.1054 ESTAB

Router# clear tcp tcb 60B75E48
[confirm]
[OK]
```

Configuring Banners Example

The following example shows how to use the **banner** global configuration commands and the **no exec-banner** line configuration command to notify your users that the server is going to be reloaded with new software:

```
! The EXEC and MOTD banners are inappropriate for the VTYS.
line vty 0 4
  no exec-banner
!
banner exec /
  This is Cisco Systems training group router.

  Unauthorized access prohibited.
  /
!
banner incoming /
  You are connected to a Hayes-compatible modem.

  Enter the appropriate AT commands.
  Remember to reset anything you have changed before disconnecting.
  /
!
banner motd /
  The router will go down at 6pm for a software upgrade
  /
```

When someone connects to the router, the MOTD banner appears before the login prompt. After the user successfully logs in to the router, the EXEC banner or incoming banner will be displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner will be displayed. For all other connections, the router will display the EXEC banner.

Setting a SLIP-PPP Banner with Banner Tokens Example

The following example sets the SLIP-PPP banner using several tokens and the percent sign (%) as the delimiting character:

```
router(config)# banner slip-ppp %
Enter TEXT message. End with the character '%'.
Starting $(encap) connection from $(gate-ip) to $(peer-ip) using a maximum packet size of
$(mtu) bytes... %
```

When you enter the **slip** command, you will see the following banner. Notice that the $\$(token)$ syntax is replaced by the corresponding configuration variable.

```
Starting SLIP connection from 192.168.69.96 to 172.16.80.8 using a maximum packet size of
1500 bytes...
```

Configuring a Menu Example

The following example allows menu users to Telnet to one of three different machines. The user can also view the output of the **show user** command and exit the menu. One hidden menu item, specified by the selection here, allows system administrators to view the current software version.

```
menu new title ^C
```

Telnet Menu

```
^C
```

```
menu new prompt ^C
```

```
Please enter your selection: ^C
```

```
menu new text 1 telnet system1
```

```
menu new command 1 telnet system1
```

```
menu new options 1 pause
```

```
menu new text 2 telnet system2
```

```
menu new command 2 telnet system2
```

```
menu new options 2 pause
```

```
menu new text b telnet systemblue
```

```
menu new command b telnet systemblue
```

```
menu new options b pause
```

```
menu new text me show user
```

```
menu new command me show user
```

```
menu new options me pause
```

```
menu new command here show version
```

```
menu new text Exit Exit
```

```
menu new command Exit menu-exit
```

```
menu new clear-screen
```

```
menu new status-line
```

```
menu new default me
```

```
menu new line-mode
```

```
!
```

