



Configuring User Profiles

This chapter describes how to configure user profiles. Information in this chapter applies to all models of the CSS, except where noted.

This chapter contains the following major sections:

- [User Profiles Overview](#)
- [User Profile Configuration Quick Start](#)
- [Configuring Idle Timeout](#)
- [Using Expert Mode](#)
- [Changing the CLI Prompt](#)
- [Modifying the History Buffer](#)
- [Configuring a Pre-Login Banner](#)
- [Configuring a Login Banner](#)
- [Copying and Saving User Profiles](#)

User Profiles Overview

The CSS contains a default-profile file that resides in the scripts directory on the CSS disk (hard disk or Flash disk). This file contains settings that are user-specific; that is, they apply uniquely to each user when that user logs in.

You can customize the following settings for each user:

- CLI prompt
- Expert mode
- History buffer
- Terminal parameters
- Login banner

Though the settings are user-specific, each default setting applies to all users until the user saves the default-profile file to a *username*-profile (where *username* is the current login username). You can continue using the default-profile file to ensure all users logging in to a CSS use the same settings. See the “[Configuring a Login Banner](#)” section for information on saving the default-profile file.

If you change a user setting and want to save this setting in the scripts directory of the current ADI, use the **copy profile** command. If you do not save this setting, the CSS stores the setting temporarily in a running profile. If you attempt to log out of the CSS without saving profile changes, the CSS prompts you that profile changes have been made and allows you to save or discard the changes.

You can also use the **save_profile** alias command to save your user-profiles settings to the scripts directory and then archive them in the CSS archive directory.

When you upgrade the ADI, the CSS does not contain all user profiles from the current ADI directory. If you wish to save user profiles permanently, use the **archive script** command. This command saves a user-profile file from the scripts directory to the archive directory. The archive directory is not overwritten during a software upgrade. Then you can restore them to the scripts directory with the **restore** command.

To access the CSS disk, FTP to the CSS. Use the appropriate commands to access the scripts directory and list the contents of the default-profile file. When logged into the CSS, use the **show profile** command to display either the default-profile file or your *username*-profile file.

For example:

```
# show profile

@prompt CSS11503
@no expert
alias all reboot "@configure;boot;rebo"
alias all shutdown "@configure;boot;shutd"
alias all logon "@configure;logging line \${LINE};exit"
alias all logoff "@configure;no logging line \${LINE};exit"
alias all aca-load "@script play service-load"
alias all dnslookup "@script play dnslookup"
alias super save_config "copy running-config startup-config;archive
startup-config"
alias super setup "script play setup"
alias super upgrade "script play upgrade"
alias super monitor "script play monitor"
alias super save_profile "copy profile user-profile;archive script
admin-profile
"
set CHECK_STARTUP_ERRORS "1" session
```

User Profile Configuration Quick Start

[Table 3-1](#) provides a quick overview of the steps required to configure a user profile. Each step includes the CLI command required to complete the task. For a complete description of each feature and all the options associated with the CLI command, see the sections following [Table 3-1](#).

Table 3-1 *User Profile Configuration Quick Start*

Task and Command Example

1. Set the length of time a session can be idle before the CSS terminates a console or Telnet session.

```
# terminal idle 15
```

2. Set the number of output lines the CLI displays on the terminal screen.

```
# terminal length 35
```

Table 3-1 User Profile Configuration Quick Start (continued)

Task and Command Example
<p>3. (Optional) Turn on the display of the --More-- prompt at the bottom of the terminal screen to indicate that additional CLI commands are to follow on the screen.</p>
<pre># terminal more</pre>
<p>4. Define how the CSS displays subnet masks in show screens.</p>
<pre># terminal netmask-format bitcount</pre>
<p>5. Set the total amount of time a session can be logged in before the CSS terminates a console or Telnet session.</p>
<pre># terminal timeout 30</pre>
<p>6. Globally set the total amount of time all sessions can be active before the CSS terminates a console or Telnet session.</p>
<pre>(config)# idle timeout 15</pre>
<p>7. (Optional) Enable expert mode to <i>disable</i> the CSS from prompting you for confirmation when you make changes. Expert mode is available in SuperUser mode and is off by default.</p>
<pre># expert</pre>
<p>8. (Optional) Change the default CLI prompt (CSS product model number followed by the # symbol).</p>
<pre>CSS11506# prompt CSS1-lab CSS1-lab#</pre>
<p>9. (Optional) Modify the CLI history buffer length.</p>
<pre># history length 80</pre>
<p>10. (Optional) Create and display a custom banner that appears after you log in to a CSS.</p>
<pre># set BANNER "mybanner" session</pre>
<p>11. Copy the running profile from the CSS to the default-profile file, an FTP server, a TFTP server, or your user-profile file. For example:</p>
<pre># copy profile default-profile</pre>
<p>Note This command is available only in SuperUser mode.</p>

Configuring User Terminal Parameters

You can control the configure the output to the system terminal screen. Use the **terminal** command to configure terminal parameters. These parameters control output to the system terminal screen. Terminal parameters are user-specific; that is, they apply uniquely to each CSS user.

Use the **copy profile user-profile** command to add terminal command parameters to your user profile so that the parameters are used each time you log in. Otherwise, you must reenter the commands for the parameters to take effect each time you log in.

This section includes the following topics:

- [Configuring Terminal Idle](#)
- [Configuring Terminal Length](#)
- [Configuring the More Terminal Prompt](#)
- [Configuring Terminal Netmask-Format](#)
- [Configuring Terminal Timeout](#)

Configuring Terminal Idle

By default, the terminal idle time is disabled. Use the **terminal idle** command to set the length of time a session can be idle before the CSS terminates a console or Telnet session. This command is available in the User and SuperUser modes. Enter an idle time between 0 and 65535 minutes. The default value is 0 (disabled).

To set a terminal idle time, enter:

```
# terminal idle 15
```

To restore the terminal idle time to the default state of disabled, enter:

```
# no terminal idle
```

Configuring Terminal Length

By default, the CSS displays 25 lines of CLI output on the terminal screen. Use the **terminal length** command to set the number of output lines the CLI displays on the terminal screen. This command is available in User and SuperUser modes. Enter the number of lines you want the CLI to display, from 2 to 65535.

To set the line number to 35, enter:

```
# terminal length 35
```

To reset the number of lines to the default of 25 lines, enter:

```
# no terminal length
```

Configuring the More Terminal Prompt

When you enter the question mark (?) character at the command line to get help about a command, the CSS displays 24 lines on the terminal. The --More-- prompt indicates that additional CLI commands are to follow. By default, the CSS disables the display of the --More-- prompt.

To display the --More-- prompt at the bottom of the terminal screen, use the **terminal more** command. Press the **Space** bar to continue viewing the next series of commands (or press the **Return** key to display only the next line). This command is available in User and SuperUser modes. The default is enabled.

You can also toggle the terminal more function on and off within a session by using the Esc-M key sequence.

To enable support for the --More-- terminal prompt, enter:

```
# terminal more
```

To disable support for the --More-- terminal prompt, enter:

```
# no terminal more
```

Configuring Terminal Netmask-Format

By default, the CSS displays dotted-decimal subnet masks in show screens. Use the **terminal netmask-format** command to determine how the CSS displays subnet masks. This command is available in User and SuperUser modes. The options for this command are as follows:

- **terminal netmask-format bitcount** - Displays masks in classless interdomain routing (CIDR) bitcount (for example, /24).
- **terminal netmask-format decimal** - Displays masks in dotted-decimal format (for example, 255.255.255.0). This is the default format.
- **terminal netmask-format hexadecimal** - Displays masks in hexadecimal format (for example, OXFFFFFF00).

For example, to display subnet masks in bit-count format, enter:

```
# terminal netmask-format bitcount
```

To restore the default display format (**decimal**), enter:

```
# no terminal netmask format
```

Configuring Terminal Timeout

By default, the CSS does not have a time limit for a console or Telnet session. Use the **terminal timeout** command to set the total amount of time a session can be logged in before the CSS terminates a console or Telnet session. This command is available in User and SuperUser modes. Enter a timeout value between 0 and 65535 minutes. The default value is 0 (disabled).

For example, to set the terminal timeout value to 30 minutes, enter:

```
# terminal timeout 30
```

To restore the terminal timeout value to the default state of disabled, enter:

```
# no terminal timeout
```

Configuring Idle Timeout

By default, the CSS does not have a idle timeout period for all active Telnet, console, FTP, SSH, and web management sessions. Use the **idle timeout** command to globally set the total amount of time all console, Telnet, SSH or FTP sessions can be active before the CSS terminates them. To set the idle timeout for all active web management sessions, use the **idle timeout web-mgmt** command.

Enter a timeout value between 0 and 65535 minutes. By default, the idle timeout is disabled (set to 0).

**Note**

To override the idle timeout value for a specific Telnet, console, SSH, or FTP session, configure the **terminal timeout command**. Terminal commands are user-specific; that is, they apply uniquely to each CSS user.

To set an idle timeout value for Telnet, console, SSH, or FTP sessions, enter:

```
(config)# idle timeout 15
```

We recommend that you configure the Telnet idle timeout for at least 30 minutes. Setting this value to 30 minutes:

- Cleans up idle Telnet sessions
- Helps prevent busy conditions due to a high number of active Telnet session

To disable the terminal timeout for Telnet, console, SSH, or FTP sessions, enter:

```
(config)# no idle timeout
```

To set an idle timeout value for all active web management sessions, enter:

```
(config)# idle timeout web-mgmt 15
```

To disable the web management timeout period, enter:

```
(config)# no idle timeout web-mgmt
```

Using Expert Mode

Expert mode allows you to turn the CSS confirmation capability on or off. Expert mode is available in SuperUser mode and is off by default. When expert mode is off, the CSS prompts you for confirmation when you:

- Execute commands that could delete or change operating parameters
- Exit a terminal session (console or Telnet) without copying the running configuration to the startup configuration
- Create services, owners, and content rules



Caution

Turning expert mode on *disables* the CSS from prompting you for confirmation when you make changes. When you exit from the CSS, all configuration changes are automatically saved to the profile and to the running-config file. You are not prompted for confirmation to save the changes.

To enable expert mode, enter:

```
# expert
```

To allow the CSS to prompt you for confirmation before executing configuration commands, enter:

```
# no expert
```

For example, when you enter the command to create an owner and expert mode is off, the CSS prompts you to verify the command, enter:

```
(config)# owner arrowpoint.com
Create owner <arrowpoint.com>, [y/n]:y
(config-owner[arrowpoint.com])#
```

Changing the CLI Prompt

The CLI default prompt appears as the CSS product model number followed by the number (#) symbol. The CSS adds a # to the prompt automatically to indicate SuperUser mode. To change the default prompt, enter the **prompt** command. For example:

```
CSS11506# prompt CSS1-1ab  
CSS1-1ab#
```

You can enter a maximum of 15 characters.

To save the new prompt, add it to your user- or default-profile file. To restore the prompt to the default, use the **no** form of the **prompt** command.

For example:

```
CSS11506# no prompt
```

Modifying the History Buffer

The history buffer stores 20 lines of the most recent CLI commands that you enter. Use the **history** command to modify the CLI history buffer length. Enter the number of lines you want in the history buffer as an integer from 0 to 256. The default is 20. This command is available only in SuperUser mode.

To set the history buffer to 80 lines, enter:

```
# history length 80
```

To disable the history function (setting of 0), enter:

```
# history length 0
```

To restore the history buffer to the default of 20 lines, enter:

```
# no history length
```

Displaying the History Buffer

Use the **show history** command to display the contents of the history buffer. The history buffer is cleared automatically upon reboot.

For example:

```
# show history

history
show history
show ip routes
show ip summary
show ip stat
clock
clock date
clock time
show history
```

Configuring a Pre-Login Banner

You can configure a custom banner that displays before you log in when you connect to a CSS. The banner is an ASCII text file that you provide and it must reside in the CSS script directory. This banner is a general banner that is the same for all users. For example, you could create a banner that includes the name of your company or a department within your company.

To configure a pre-login banner, use the **prelogin-banner** command in global configuration mode. This command has the following syntax:

```
prelogin-banner "filename"
```

The *filename* variable is the name of the ASCII text file that contains the pre-login banner text. Enter a quoted text string with a maximum of 32 characters.

For example, to configure a pre-login banner file called newBanner:

1. Use any text editor (for example, Notepad or Wordpad) to create a custom banner called newBanner and save it as a text file. The maximum line width is 80 characters.
2. FTP the text file to the CSS script directory as follows:
 - a. From the directory that contains the banner text file, FTP to the CSS. For example, enter: **ftp 192.168.12.5**.
 - b. At the FTP prompt, log in to the CSS.
 - c. Enter **cd script** to change to the CSS script directory.
 - d. Enter **put newBanner newBanner**.
FTP transfers the banner file to the CSS script directory.
3. To complete the configuration, enter the following command at the CSS CLI:

```
(config)# prelogin-banner newBanner
```

The next time you connect to the CSS, the custom banner appears.

To reset the default behavior of the CSS to no pre-login banner, enter:

```
(config)# no prelogin-banner
```

Configuring a Login Banner

The CSS banner is an ASCII text file that you provide and it must reside in the CSS /script directory. Because this feature is part of your user profile, you can have your own custom banner associated with your login name.

Use the **set** command with the BANNER environment variable to create a custom banner that appears after you log in to a CSS. To configure a login banner:

1. Use any text editor (for example, Notepad or Wordpad) to create a custom banner and save it as a text file in the CSS script directory. The maximum line width is 80 characters.
2. FTP the banner file to the CSS script directory. See the [“Configuring a Pre-Login Banner”](#) section.

3. At the CSS CLI, set the environment variable BANNER and point to the mybanner file:

```
# set BANNER "mybanner" session
```

The keyword **session** causes the CSS to create the environment variable every time you log in.

4. To complete the configuration, enter the following command:

```
# copy profile user-profile
```

The next time you log in to the CSS, your custom banner appears.

To provide the same banner for all users, replace the command in Step 4 with:

```
# copy profile default-profile
```

Copying and Saving User Profiles

To copy the running profile from the CSS to the default-profile file, an FTP server, a TFTP server, or your user-profile file, use the **copy profile** command. This command is available only in SuperUser mode.

If you exit the CSS without copying changes in the running profile to your *username*-profile or default-profile file, the CSS prompts you that the profile has changed and queries whether you want to save your changes. If you respond with **y**, the CSS copies the running profile to your *username*-profile or the default-profile.

This section includes the following topics:

- [Copying the Running Profile to the Default-Profile](#)
- [Copying the Running Profile to a User Profile](#)
- [Copying the Running Profile to an FTP Server](#)
- [Copying the Running Profile to a TFTP Server](#)

Copying the Running Profile to the Default-Profile

To copy the running profile to the default profile, use the **copy profile default-profile** command. This command is available only in SuperUser mode.

For example:

```
# copy profile default-profile
```

Copying the Running Profile to a User Profile

To copy the changes made to the running profile to the user profile, use the **copy profile user-profile** command. This command is available only in SuperUser mode. This command creates a file *username-profile* if one does not exist (where *username* is the current username).

For example:

```
# copy profile user-profile
```

Copying the Running Profile to an FTP Server

To copy the running profile to an FTP server, use the **copy profile ftp** command. This command is available only in SuperUser mode. The syntax is:

```
copy profile ftp ftp_record filename
```

The variables for this command are as follows:

- *ftp_record* - The name of the FTP record file that contains the server IP address, username, and password. Enter an unquoted text string with no spaces and a maximum of 32 characters.
- *filename* - The name you want to assign to the file on the server. Include the full path to the file. Enter an unquoted text string with no spaces.

For example:

```
# copy profile ftp arrowrecord \records\arrowftprecord
```

Copying the Running Profile to a TFTP Server

To copy the running profile to a TFTP server, use the **copy profile tftp** command. This command is available only in SuperUser mode. The syntax is:

```
copy profile tftp ip_or_host filename
```

The variables for this command are as follows:

- *ip_or_host* - The IP address or host name of the server to receive the file. Enter an IP address in dotted-decimal notation (for example, 192.168.11.1) or in mnemonic host-name format (for example, myhost.mydomain.com).
- *filename* - The name you want to assign to the file on the server. Include the full path to the file. Enter an unquoted text string with no spaces and a maximum of 32 characters.

For example:

```
# copy profile tftp 192.168.3.6 \home\bobo\bobo-profile
```

Where to Go Next

[Chapter 4, Using the CSS Logging Features](#), describes how to enable logging, set up the log buffer, and determine where to send the activity information. This chapter also provides information on interpreting sys.log messages and a description of frequently queried messages.

■ Where to Go Next