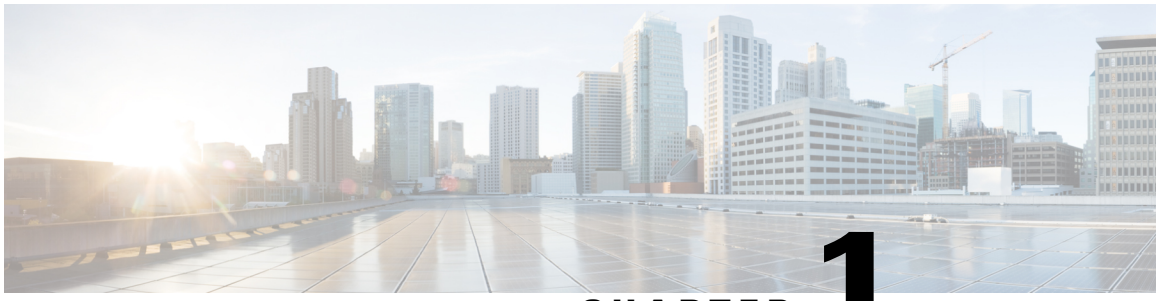
User Security Configuration Guide, Cisco IOS Release 15E

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

Image Verification

The Image Verification feature allows users to automatically verify the integrity of Cisco IOS images. Thus, users can be sure that the image is protected from accidental corruption, which can occur at any time during transit, starting from the moment the files are generated by Cisco until they reach the user. The efficiency of Cisco IOS routers is also improved because the routers can now automatically detect when the integrity of an image is accidentally corrupted as a result of transmission errors or disk corruption.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Image Verification

Cisco IOS Release 12.2(18)S and 12.0(26)S Only

Image Verification is applied to and attempted on any file; however, if the file is not an image file, image verification will not occur and you will see the following error, “SIGNATURE-NOT-FOUND.”

Cisco IOS Release 12.3(4)T Only

Image Verification is applied only to image files. If any other file type is copied or verified, you will not receive a warning that image verification did occur, and the command (copy or verify) will silently succeed.

Note The Image Verification feature can only be used to check the integrity of a Cisco IOS software image that is stored on a Cisco IOS device. It cannot be used to check the integrity of an image on a remote file system or an image running in memory.

Information About Image Verification

How Image Verification Works

Because a production image undergoes a sequence of transfers before it is copied into the memory of a router, the integrity of the image is at risk of accidental corruption every time a transfer occurs. When downloading an image from Cisco.com, a user can run a message-digest5 (MD5) hash on the downloaded image and verify that the MD5 digest posted on Cisco.com is the same as the MD5 digest that is computed on the user's server. However, many users choose not to run an MD5 digest because it is 128-bits long and the verification is manual. Image verification allows the user to automatically validate the integrity of all downloaded images, thereby, significantly reducing user interaction.

How to Use Image Verification

Globally Verifying the Integrity of an Image

The **file verify auto** command enables image verification globally; that is, all images that are to be copied (via the **copy** command) or reloaded (via the **reload** command) are automatically verified. Although both the **copy** and **reload** commands have a **/verify** keyword that enables image verification, you must issue the keyword each time you want to copy or reload an image. The **file verify auto** command enables image verification by default, so you no longer have to specify image verification multiple times.

If you have enabled image verification by default but prefer to disable verification for a specific image copy or reload, the **/noverify** keyword, along with either the **copy** or the **reload** command, will override the **file verify auto** command.

Use this task to enable automatic image verification.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **file verify auto**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**file verify auto** **Example:** `Router(config)# file verify auto`	Enables automatic image verification.
Step 4	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode. You must exit global configuration mode if you are going to copy or reload an image.

What to Do Next

After issuing the **file verify auto** command, you do not have to issue the **/verify** keyword with the **copy** or the **reload** command because each image that is copied or reloaded will be automatically verified.

Verifying the Integrity of an Image That Is About to Be Copied

When issuing the **copy** command, you can verify the integrity of the copied file by entering the **/verify** keyword. If the integrity check fails, the copied file will be deleted. If the file that is about to be copied does not have an embedded hash (an old image), you will be prompted whether or not to continue with the copying process. If you choose to continue, the file will be successfully copied; if you choose not to continue, the copied file will be deleted.

Without the **/verify** keyword, the **copy** command could copy a file that is not valid. Thus, after the **copy** command has been successfully executed, you can issue the **verify** command at any time to check the integrity of the files that are in the storage of the router.

Use this task to verify the integrity of an image before it is copied onto a router.

SUMMARY STEPS

1. **enable**
2. **copy** [**/erase**] [**/verify**| **/noverify**] *source-url destination-url*
3. **verify** [**/md5** [*md5-value*]] *filesystem: file-url*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**copy** [**/erase**] [**/verify**| **/noverify**] *source-url destination-url* **Example:** `Router# copy /verify tftp://10.1.1.1/jdoe/c7200-js-mz disk0:`	Copies any file from a source to a destination. • **/verify** --Verifies the signature of the destination file. If verification fails, the file will be deleted. • **/noverify** --Does not verify the signature of the destination file before the image is copied. **Note** **/noverify** is often issued if the **file verify auto** command is enabled, which automatically verifies the signature of all images that are copied.
Step 3	**verify** [**/md5** [*md5-value*]] *filesystem: file-url*] **Example:** `Router# verify bootflash://c7200-kboot-mz.121-8a.E`	(Optional) Verifies the integrity of the images in the router's storage.

Verifying the Integrity of an Image That Is About to Be Reloaded

By issuing the **reload** command with the /**verify** keyword, the image that is about to be loaded onto your system will be checked for integrity. If the /**verify** keyword is specified, image verification will occur before the system initiates the reboot. Thus, if verification fails, the image will not be loaded.

Note Because different platforms obtain the file that is to be loaded in various ways, the file specified in BOOTVAR will be verified. If a file is not specified, the first file on each subsystem will be verified. On certain platforms, because of variables such as the configuration register, the file that is verified may not be the file that is loaded.

Use this task to verify the integrity of an image before it is reloaded onto a router.

SUMMARY STEPS

1. **enable**
2. **reload** [[**warm**] [**/verify**| **/noverify**] *text* | [**warm**] [**/verify**| **/noverify**] **in** [*hh* **:** *mm* [*text*] | [**warm**] [**/verify**| **/noverify**] **at** *hh* **:** *mm* [*month day* | *day month*] [*text*] | [**warm**] [**/verify**| **/noverify**] **cancel**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**reload** [[**warm**] [**/verify**| **/noverify**] *text*	[**warm**] [**/verify**| **/noverify**] **in** [*hh* **:** *mm* [*text*]	[**warm**] [**/verify**| **/noverify**] **at** *hh* **:** *mm* [*month day*	*day month*] [*text*]	[**warm**] [**/verify**| **/noverify**] **cancel**] **Example:** `Router# reload /verify`	Reloads the operating system. • /**verify**--Verifies the signature of the destination file. If verification fails, the file will be deleted. • **/noverify** --Does not verify the signature of the destination file before the image is reloaded. **Note** **/noverify** is often issued if the **file verify auto** command is enabled, which automatically verifies the signature of all images that are copied.

Configuration Examples for Image Verification

Global Image Verification Example

The following example shows how to enable automatic image verification. After enabling this command, image verification will automatically occur for all images that are either copied (via the **copy** command) or reloaded (via the **reload** command).

```
Router(config)# file verify auto
```

Image Verification via the copy Command Example

The following example shows how to specify image verification before copying an image:

```
Router# copy /verify tftp://10.1.1.1/jdoe/c7200-js-mz disk0:
Destination filename [c7200-js-mz]?
Accessing tftp://10.1.1.1/jdoe/c7200-js-mz...
Loading jdoe/c7200-js-mz from 10.1.1.1 (via FastEthernet0/0):!!!!!!!!!!!!!!!
```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 19879944 bytes]
19879944 bytes copied in 108.632 secs (183003 bytes/sec)
Verifying file integrity of disk0:/c7200-js-mz
.....................................................................................
.....................................................................................
.....................................................................................
.........................Done!
Embedded Hash   MD5 :CFA258948C4ECE52085DCF428A426DCD
Computed Hash       MD5 :CFA258948C4ECE52085DCF428A426DCD
CCO Hash               MD5 :44A7B9BDDD9638128C35528466318183
Signature Verified
```

Image Verification via the reload Command Example

The following example shows how to specify image verification before reloading an image onto the router:

```
Router# reload /verify
Verifying file integrity of bootflash:c7200-kboot-mz.121-8a.E
%ERROR:Signature not found in file bootflash:c7200-kboot-mz.121-8a.E.
Signature not present. Proceed with verify? [confirm]
Verifying file disk0:c7200-js-mz
.....................................................................
.....................................................Done!
Embedded Hash     MD5 :CFA258948C4ECE52085DCF428A426DCD
Computed Hash      MD5 :CFA258948C4ECE52085DCF428A426DCD
CCO Hash          MD5 :44A7B9BDDD9638128C35528466318183
Signature Verified
Proceed with reload? [confirm]n
```

Verify Command Sample Output Example

The following example shows how to specify image verification via the **verify** command:

```
Router# verify disk0:c7200-js-mz
%Filesystem does not support verify operations
Verifying file integrity of disk0:c7200-js-mz.....................................
.....................................................................................Done!
Embedded Hash     MD5 :CFA258948C4ECE52085DCF428A426DCD
Computed Hash      MD5 :CFA258948C4ECE52085DCF428A426DCD
CCO Hash          MD5 :44A7B9BDDD9638128C35528466318183
```

Signature Verified

Additional References

Related Documents

Related Topic	Document Title
Configuration tasks and information for loading, maintaining, and rebooting system images	Using the Cisco IOS Integrated File System feature module in the *Cisco IOS Configuration Fundamentals and Network Management Configuration Guide* , Release 12.4T.

Related Topic	Document Title
Additional commands for loading, maintaining, and rebooting system images	*Cisco IOS Configuration Fundamentals Command Reference* , Release 12.4T

Standards

Standard	Title
None	--

MIBs

MIB	MIBs Link
• None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
None	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

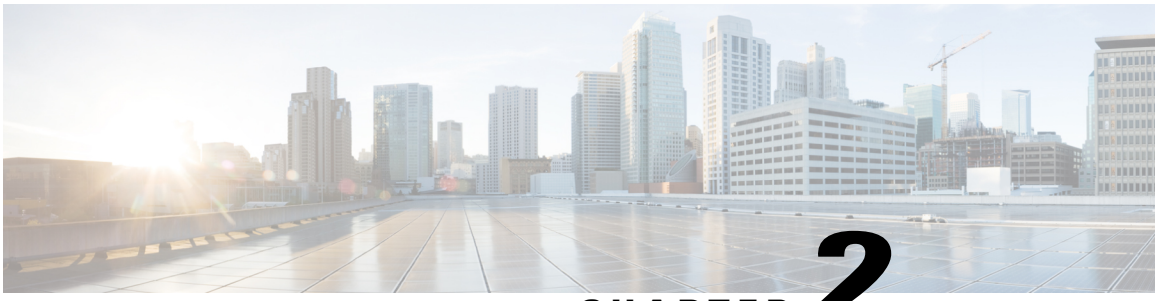
Feature Information for Image Verification

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Image Verification

Feature Name	Releases	Feature Information
Image Verification	Cisco IOS 15.0(2)EX	The Image Verification feature allows users to automatically verify the integrity of Cisco IOS images. In Cisco IOS Release 15.0(2)EX, this feature was supported on the following platforms: • Catalyst 2960-S Series Switches The following commands were introduced or modified: **copy**, **file verify auto**, **reload**, **verify**.



CHAPTER 2

Role-Based CLI Access

The Role-Based CLI Access feature allows the network administrator to define views, which are a set of operational commands and configuration capabilities that provide selective or partial access to Cisco IOS EXEC and configuration (config) mode commands. Views restrict user access to Cisco IOS command-line interface (CLI) and configuration information; that is, a view can define what commands are accepted and what configuration information is visible. Thus, network administrators can exercise better control over access to Cisco networking devices.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Role-Based CLI Access

Your image must support CLI views.

Restrictions for Role-Based CLI Access

Lawful Intercept Images Limitation

CLI views are a part of all platforms and Cisco IOS images because they are a part of the Cisco IOS parser. However, the lawful intercept view is available only in images that contain the lawful intercept subsystem.

Maximum Number of Allowed Views

The maximum number of CLI views and superviews, including one lawful intercept view, that can be configured is 15. (This does not include the root view.)

Information About Role-Based CLI Access

Benefits of Using CLI Views

Although users can control CLI access via both privilege levels and enable mode passwords, these functions do not provide network administrators with the necessary level of detail needed when working with Cisco IOS devices. CLI views provide a more detailed access control capability for network administrators, thereby, improving the overall security and accountability of Cisco IOS software.

As of Cisco IOS Release 12.3(11)T, network administrators can also specify an interface or a group of interfaces to a view; thereby, allowing access on the basis of specified interfaces.

Root View

When a system is in root view, it has all of the access privileges as a user who has level 15 privileges. If the administrator wishes to configure any view to the system (such as a CLI view, a superview, or a lawful intercept view), the system must be in root view.

The difference between a user who has level 15 privileges and a root view user is that a root view user can configure a new view and add or remove commands from the view. Also, when you are in a CLI view, you have access only to the commands that have been added to that view by the root view user.

Lawful Intercept View

Like a CLI view, a lawful intercept view restricts access to specified commands and configuration information. Specifically, a lawful intercept view allows a user to secure access to lawful intercept commands that are held within the TAP-MIB, which is a special set of simple network management protocol (SNMP) commands that store information about calls and users.

Commands available in lawful intercept view belong to one of the these categories:

- Lawful intercept commands that should not be made available to any other view or privilege level
- CLI views that are useful for lawful intercept users but do not have to be excluded from other views or privilege levels

Superview

A superview consists of one or more CLI views, which allow users to define what commands are accepted and what configuration information is visible. Superviews allow a network administrator to easily assign all users within configured CLI views to a superview instead of having to assign multiple CLI views to a group of users.

Superviews contain these characteristics:

- A CLI view can be shared among multiple superviews.
- Commands cannot be configured for a superview; that is, you must add commands to the CLI view and add that CLI view to the superview.
- Users who are logged into a superview can access all of the commands that are configured for any of the CLI views that are part of the superview.
- Each superview has a password that is used to switch between superviews or from a CLI view to a superview.
- If a superview is deleted, its associated CLI views are not deleted.

View Authentication via a New AAA Attribute

View authentication is performed by an external authentication, authorization, and accounting (AAA) server via the new attribute **cli-view-name**.

AAA authentication associates only one view name to a particular user; that is, only one view name can be configured for a user in an authentication server.

How to Use Role-Based CLI Access

Configuring a CLI View

Perform this task to create a CLI view and add commands or interfaces to the view, as appropriate.

Before You Begin

Before you create a view, you must perform the following tasks:

- Enable AAA via the **aaa new-model** command .
- Ensure that your system is in root view-not privilege level 15.

SUMMARY STEPS

1. **enable view**
2. **configure terminal**
3. **parser view** *view-name* [**inclusive**]
4. **secret** [**0** | **5**] *encrypted-password*
5. **commands** *parser-mode* {**exclude** | **include-exclusive** | **include**} [**all**] [**interface** *interface-name* | *command*]
6. **end**
7. **enable** [*privilege-level* | **view** *view-name*]
8. **show parser view all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable view** **Example:** `Device> enable view`	Enables root view. • Enter your privilege level 15 password (for example, root password) if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**parser view** *view-name* [**inclusive**]
	
	Example:
	`Device(config)# parser view first inclusive`
	`Device(config-view)#`
		Creates a view including all commands by default. If the **inclusive** keyword option is not selected, it creates a view excluding all commands by default. You are in the view configuration mode.

Step 4	**secret** [**0**	**5**] *encrypted-password* **Example:** `Device(config-view)# secret 5 secret`	Associates a CLI view or superview with a password. **Note** You must issue this command before you can configure additional attributes for the view. **Note** With CSCts50236, the password can be removed or overwritten. Use the **no secret** command to remove the configured password.
Step 5	**commands** *parser-mode* {**exclude**	**include-exclusive**	**include**} [**all**] [**interface** *interface-name*	*command*] **Example:** `Device(config-view)# commands exec include show version`	Adds commands or interfaces to a view and specifies the mode in which the specified command exists.

	Command or Action	Purpose
Step 6	**end** **Example:** `Device(config-view)# end`	Exits view configuration mode and returns to privileged EXEC mode.
Step 7	**enable** [*privilege-level* | **view** *view-name*] **Example:** `Device# enable view first`	Prompts you for a password to access a configured CLI view, and you can switch from one view to another view. Enter the password to access the CLI view.
Step 8	**show parser view all** **Example:** `Device# show parser view all`	(Optional) Displays information for all views that are configured on the device. **Note** Although this command is available for both root and lawful intercept users, the **all** keyword is available only to root users. However, the **all** keyword can be configured by a user in root view to be available for users in lawful intercept view and CLI view.

Troubleshooting Tips

You must associate a password with a view. If you do not associate a password, and you attempt to add commands to the view using the **commands** command, a system message such as the following is displayed:

```
%Password not set for view <viewname>.
```

Configuring a Lawful Intercept View

Perform this task to initialize and configure a view for lawful-intercept-specific commands and configuration information.

Before You Begin

Before you initialize a lawful intercept view, ensure that the privilege level is set to 15 using the **privilege** command.

Note Only an administrator or a user who has level 15 privileges can initialize a lawful intercept view.

SUMMARY STEPS

1. **enable view**
2. **configure terminal**
3. **li-view** *li-password* **user** *username* **password** *password*
4. **username lawful-intercept** [*name*] [**privilege** *privilege-level* | **view** *view-name*] **password** *password*
5. **parser view** *view-name*
6. **secret 5** *encrypted-password*
7. **name** *new-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable view** **Example:** `Device> enable view`	Enables root view. • Enter your privilege level 15 password (for example, root password) if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**li-view** *li-password* **user** *username* **password** *password* **Example:** `Device(config)# li-view lipass user li_admin password li_adminpass`	Initializes a lawful intercept view. After the li-view is initialized, you must specify at least one user via **user** *username* **password** *password* options.
Step 4	**username lawful-intercept** [*name*] [**privilege** *privilege-level*	**view** *view-name*] **password** *password* **Example:** `Device(config)# username lawful-intercept li-user1 password li-user1pass`	Configures lawful intercept users on a Cisco device.
Step 5	**parser view** *view-name* **Example:** `Device(config)# parser view li view name`	(Optional) Enters view configuration mode, which allows you to change the lawful intercept view password or the lawful intercept view name.
Step 6	**secret 5** *encrypted-password* **Example:** `Device(config-view)# secret 5 secret`	(Optional) Changes an existing password for a lawful intercept view.

	Command or Action	Purpose
Step 7	**name** *new-name* **Example:** `Device(config-view)# name second`	(Optional) Changes the name of a lawful intercept view. If this command is not issued, the default name of the lawful intercept view is "li-view."

Troubleshooting Tips

To display information for all users who have access to a lawful intercept view, issue the **show users lawful-intercept** command. (This command is available only to authorized lawful intercept view users.)

Configuring a Superview

Perform this task to create a superview and add at least one CLI view to the superview.

Before You Begin

Before adding a CLI view to a superview, ensure that the CLI views that are added to the superview are valid views in the system; that is, the views have been successfully created using the **parser view** command.

Note You can add a view to a superview only after you configure a password for the superview (using the **secret 5** command). Thereafter, issue the **view** command in view configuration mode to add at least one CLI view to the superview.

SUMMARY STEPS

1. **enable view**
2. **configure terminal**
3. **parser view** *superview-name* **superview**
4. **secret 5** *encrypted-password*
5. **view** *view-name*
6. **end**
7. **show parser view all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable view**	Enables root view.

	Command or Action	Purpose
	Example: `Device> enable view`	• Enter your privilege level 15 password (for example, root password) if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**parser view** *superview-name* **superview** **Example:** `Device(config)# parser view su_view1 superview`	Creates a superview and enters view configuration mode.
Step 4	**secret 5** *encrypted-password* **Example:** `Device(config-view)# secret 5 secret`	Associates a CLI view or superview with a password. **Note** You must issue this command before you can configure additional attributes for the view.
Step 5	**view** *view-name* **Example:** `Device(config-view)# view view_three`	Adds a normal CLI view to a superview. Issue this command for each CLI view that is to be added to a given superview.

Step 6	**end**
	Example:
	`Device(config-view)# end`
	`Device#`
		Exits view configuration mode and returns to privileged EXEC mode.

Step 7	**show parser view all** **Example:** `Device# show parser view`	(Optional) Displays information for all views that are configured on the device. **Note** Although this command is available for both root and lawful intercept users, the **all** keyword is available only to root users. However, the **all** keyword can be configured by a user in root view to be available for users in lawful intercept view and CLI view.

Monitoring Views and View Users

To display debug messages for all views-root, CLI, lawful intercept, and superview-use the **debug parser view** command in privileged EXEC mode.

Configuration Examples for Role-Based CLI Access

Example: Configuring a CLI View

The following example shows how to configure two CLI views, “first” and “second". Thereafter, you can verify the CLI view in the running configuration.

```
Device(config)# parser view first inclusive
Device(config-view)# secret 5 firstpass
Device(config-view)# command exec exclude show version
Device(config-view)# command exec exclude configure terminal
Device(config-view)# command exec exclude all show ip
Device(config-view)# exit
Device(config)# parser view second
Device(config-view)# secret 5 secondpass
Device(config-view)# command exec include-exclusive show ip interface
Device(config-view)# command exec include logout
Device(config-view)# exit
!
!
Device(config-view)# do show running-config | beg view

parser view first inclusive
 secret 5 $1$MCmh$QuZaU8PIMPlff9sFCZvgW/
 commands exec exclude configure terminal
 commands exec exclude configure
 commands exec exclude all show ip
 commands exec exclude show version
 commands exec exclude show
!
parser view second
 secret 5 $1$iP2M$R16BXKecMEiQesxLyqygW.
 commands exec include-exclusive show ip interface
 commands exec include show ip
 commands exec include show
 commands exec include logout
!
```

Example: Verifying a CLI View

After you have configured the CLI views “first” and “second", you can issue the **enable view**command to verify which commands are available in each view. The following example shows which commands are available inside the CLI view “first” after the user has logged into this view. (Because the **show ip** command is configured with the all option, a complete set of suboptions is shown, except the **show ip interface** command, which is using the **include-exclusive** keyword in the second view.)

```
Device# enable view first
Password:
Device# ?
Exec commands:
```

```
  configure  Enter configuration mode
  enable     Turn on privileged commands
  exit       Exit from the EXEC
  show       Show running system information
Device# show ?
  ip       IP information
  parser   Display parser information
  version  System hardware and software status
Device# show ip ?

  access-lists            List IP access lists
  accounting              The active IP accounting database
  aliases                 IP alias table
  arp                     IP ARP table
  as-path-access-list     List AS path access lists
  bgp                     BGP information
  cache                   IP fast-switching route cache
  casa                    display casa information
  cef                     Cisco Express Forwarding
  community-list          List community-list
  dfp                     DFP information
  dhcp                    Show items in the DHCP database
  drp                     Director response protocol
  dvmrp                   DVMRP information
  eigrp                   IP-EIGRP show commands
  extcommunity-list       List extended-community list
  flow                    NetFlow switching
  helper-address          helper-address table
  http                    HTTP information
  igmp                    IGMP information
  irdp                    ICMP Device Discovery Protocol
.
.
.
```

Example: Configuring a Lawful Intercept View

The following example shows how to configure a lawful intercept view, add users to the view, and verify the users that were added:

```
!Initialize the LI-View.
Device(config)# li-view lipass user li_admin password li_adminpass
Device(config)# end
! Enter the LI-View; that is, check to see what commands are available within the view.
Device# enable view li-view
Password:
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# parser view li-view

Device(config-view)# ?
View commands:
  commands  Configure commands for a view
  default   Set a command to its defaults
  exit      Exit from view configuration mode
  name      New LI-View name
  no        Negate a command or set its defaults
  password  Set a password associated with CLI views
Device(config-view)#
! NOTE:LI View configurations are never shown as part of 'running-configuration'.
! Configure LI Users.
Device(config)# username lawful-intercept li-user1 password li-user1pass

Device(config)# username lawful-intercept li-user2 password li-user2pass
! Displaying LI User information.
Device# show users lawful-intercept
li_admin
li-user1
```

```
li-user2
Device#
```

Note The lawful intercept view is available only on specific images and the view name option is available only in the LI view.

Example: Configuring a Superview

The following sample output from the **show running-config** command shows that “view_one” and “view_two” have been added to superview “su_view1", “view_three”, and “view_four” have been added to superview “su_view2”:

```
Device# show running-config
!
parser view su_view1 superview
 secret 5 <encoded password>
 view view_one
 view view_two
!
parser view su_view2 superview
 secret 5 <encoded password>
 view view_three
 view view_four
!
```

Additional References for Role-Based CLI Access

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Security commands	• Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
SNMP, MIBs, CLI configuration	*Cisco IOS Network Management Configuration Guide* , Release 15.0.
Privilege levels	"Configuring Security with Passwords, Privileges and Logins" module.

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Role-Based CLI Access

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 2: Feature Information for Role-Based CLI Access

Feature Name	Releases	Feature Information
Role-Based CLI Access	Cisco IOS 15.0(2)SG	The Role-Based CLI Access feature enables network administrators to restrict user access to CLI and configuration information. The CLI view capability was extended to restrict user access on a per-interface level, and additional CLI views were introduced to support the extended view capability. Also, support to group configured CLI views into a superview was introduced. The following commands were introduced or modified: **commands (view)**, **enable**, **li-view**, **name (view)**, **parser view , parser view superview**, **secret**, **show parser view**, **show users**, **username**, and **view**.

Feature Name	Releases	Feature Information
Role-Based CLI Inclusive Views	Cisco IOS 15.2(2)E	The Role-Based CLI Inclusive Views feature enables a standard CLI view including all commands by default. In Cisco IOS Release 15.2(2)E, this feature was supported on the following platforms: • Catalyst 2960-X Series Switches • Catalyst 2960-S Series Switches • Catalyst 2960-C Series Switches • Catalyst 3750-X Series Switches • Catalyst 3560-X Series Switches • Catalyst 4500E Supervisor Engine 6-E The following command was modified: **parser view inclusive**.