Cisco 4000 Series ISRs Software Configuration Guide, Cisco IOS XE 17

First Published: 2019-11-14

Last Modified: 2024-08-26

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

Text Part Number: OL-29328-03

CONTENTS

Full Cisco Trademarks with Software License

Preface

Preface

This section briefly describes the objectives of this document and provides links to additional information on related products and services:

Preface

This preface describes the audience, organization, and conventions of this document. It also provides information on how to obtain other documentation.

This preface includes the following sections:

Audience and Scope

This document is designed for the person who is responsible for configuring your Cisco Enterprise router. This document is intended primarily for the following audiences:

- Customers with technical networking background and experience.
- System administrators familiar with the fundamentals of router-based internetworking but who might not be familiar with Cisco IOS software.
- System administrators who are responsible for installing and configuring internetworking equipment, and who are familiar with Cisco IOS software.

Feature Compatibility

For more information about the Cisco IOS XE software, including features available on your device as described in the configuration guides, see the respective router documentation set.

To verify support for specific features, use the Cisco Feature Navigator tool. This tool enables you to determine the Cisco IOS XE software images that support a specific software release, feature set, or a platform.

Document Conventions

This documentation uses the following conventions:

Convention	Description
^ or **Ctrl**	The ^ and **Ctrl** symbols represent the Control key. For example, the key combination ^**D** or **Ctrl-D** means hold down the **Control** key while you press the **D** key. Keys are indicated in capital letters but are not case sensitive.
string	A string is a nonquoted set of characters shown in italics. For example, when setting an SNMP community string to public, do not use quotation marks around the string or the string will include the quotation marks.

The command syntax descriptions use the following conventions:

Convention	Description
bold	Bold text indicates commands and keywords that you enter exactly as shown.
italics	Italic text indicates arguments for which you supply values.
[x]	Square brackets enclose an optional element (keyword or argument).
|	A vertical line indicates a choice within an optional or required set of keywords or arguments.
[x | y]	Square brackets enclosing keywords or arguments separated by a vertical line indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a vertical line indicate a required choice.

Nested sets of square brackets or braces indicate optional or required choices within optional or required elements. For example, see the following table.

Convention	Description
[x {y | z}]	Braces and a vertical line within square brackets indicate a required choice within an optional element.

Examples use the following conventions:

Convention	Description
`screen`	Examples of information displayed on the screen are set in Courier font.
`bold screen`	Examples of text that you must enter are set in Courier bold font.
< >	Angle brackets enclose text that is not printed to the screen, such as passwords.
!	An exclamation point at the beginning of a line indicates a comment line. Exclamation points are also displayed by the Cisco IOS XE software for certain processes.
[ ]	Square brackets enclose default responses to system prompts.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Note Means *reader take note*. Notes contain helpful suggestions or references to materials that may not be contained in this manual.

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco DevNet.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Documentation Feedback

To provide feedback about Cisco technical documentation, use the feedback form available in the right pane of every online document.

Troubleshooting

For the most up-to-date, detailed troubleshooting information, see the Cisco TAC website at https://www.cisco.com/en/US/support/index.html.

Go to **Products by Category** and choose your product from the list, or enter the name of your product. Look under **Troubleshoot and Alerts** to find information for the issue that you are experiencing.

CHAPTER 1

Overview

This document is a summary of software functionality that is specific to the Cisco 4000 Series Integrated Services Routers (ISRs).

The following table lists the router models that belong to the Cisco 4000 Series ISRs.

Table 1: Cisco 4000 Series Router Models

Cisco 4400 Series ISRs	Cisco 4300 Series ISR s	Cisco 4200 Series ISRs
• Cisco 4431 ISR • Cisco 4451 ISR • Cisco 4461 ISR	• Cisco 4321 ISR • Cisco 4331 ISR • Cisco 4351 ISR	Cisco 4221 ISR

Note Unless otherwise specified, the information in this document is applicable to both Cisco 4400 Series, Cisco 4300 Series and Cisco 4200 Series routers.

The following sections are included in this chapter:

Introduction

The Cisco 4000 Series ISRs are modular routers with LAN and WAN connections that can be configured by means of interface modules, including Cisco Enhanced Service Modules (SM-Xs), and Network Interface Modules (NIMs). NIM slots also support removable storage for hosted applications.

The following features are provided for enterprise and service provider applications:

- Enterprise Applications
  - High-end branch gateway
  - Regional site aggregation
  - Key server or PfR primary controller

- Device consolidation or "Rack in a Box"
- Service Provider Applications
  - High-end managed services in Customer-Premises Equipment (CPE)
  - Services consolidation platform
  - Route reflector or shadow router
  - Flexible customer edge router

The router runs Cisco IOS XE software, and uses software components in many separate processes. This modular architecture increases network resiliency, compared to standard Cisco IOS software.

Processes

The list of background processes in the following table may be useful for checking router state and troubleshooting. However, you do not need to understand these processes to understand most router operations.

Table 2: Individual Processes

Process	Purpose	Affected FRUs	Sub Package Mapping
Chassis Manager	Controls chassis management functions, including management of the High Availability (HA) state, environmental monitoring, and FRU state control.	RP SIP ESP	RPControl SIPBase ESPBase
Host Manager	Provides an interface between the IOS process and many of the information gathering functions of the underlying platform kernel and operating system.	RP SIP ESP	RPControl SIPBase ESPBase
Logger	Provides IOS logging services to processes running on each FRU.	RP SIP ESP	RPControl SIPBase ESPBase
IOS	Implements all forwarding and routing features for the router.	RP	RPIOS

Process	Purpose	Affected FRUs	Sub Package Mapping
Forwarding Manager	Manages downloading of configuration details to the ESP and the communication of forwarding plane information, such as statistics, to the IOS process.	RP ESP	RPControl ESPBase
Pluggable Services	Provide integration between platform policy applications, such as authentication and the IOS process.	RP	RPControl
Shell Manager	Provides user interface (UI) features relating to non-IOS components of the consolidated package. These features are also available for use in diagnostic mode when the IOS process fails.	RP	RPControl
IO Module process	Exchanges configuration and other control messages with a NIM, or Enhanced Service Module (SM-X).	IO Module	SIPSPA
CPP driver process	Manages CPP hardware forwarding engine on the ESP.	ESP	ESPBase
CPP HA process	Manages HA state for the CPP hardware forwarding engine.	ESP	ESPBase
CPP SP process	Performs high-latency tasks for the CPP-facing functionality in the ESP instance of the Forwarding Manager process.	ESP	ESPBase

For further details of router capabilities and models, see the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

CHAPTER 2

Configure Initial Router Settings on Cisco 4000 Series ISRs

This chapter describes how to perform the initial configuration on Cisco 4000 Series Integrated Services Routers (ISRs). It contains the following sections:

Perform Initial Configuration on Cisco 4000 Series ISRs

You can perform initial configuration on Cisco 4000 Series ISRs by using either the setup command facility or the Cisco IOS command-line interface (CLI):

Use Cisco Setup Command Facility

The setup command facility prompts you to enter the information about your router and network. The facility steps guides you through the initial configuration, which includes LAN and WAN interfaces. For more general information about the setup command facility, see the following document:

Cisco IOS Configuration Fundamentals Configuration Guide , Release 12.4, Part 2: Cisco IOS User Interfaces: Using AutoInstall and Setup: http://www.cisco.com/c/en/us/support/ios-nx-os-software/ios-xe-3s/products-installation-and-configuration-guides-list.html.

This section explains how to configure a hostname for the router, set passwords, and configure an interface to communicate with the management network.

Note The messages that are displayed will vary based on your router model, the installed interface modules, and the software image. The following example and the user entries (in **bold**) are shown only as examples.

Note If you make a mistake while using the setup command facility, you can exit and run the setup command facility again. Press **Ctrl-C**, and enter the **setup** command in privileged EXEC mode (Router#)

To configure the initial router settings by using the setup command facility, follow these steps:

SUMMARY STEPS

1. From the Cisco IOS-XE CLI, enter the **setup** command in privileged EXEC mode:
2. To proceed using the setup command facility, enter **yes**.
3. To enter the basic management setup, enter **yes**.
4. Enter a hostname for the router (this example uses ‘myrouter’):
5. Enter an enable secret password. This password is encrypted (for more security) and cannot be seen when viewing the configuration.
6. Enter an enable password that is different from the enable secret password. This password is *not* encrypted (and is less secure) and can be seen when viewing the configuration.
7. Enter the virtual terminal password, which prevents unauthenticated access to the router through ports other than the console port:
8. Respond to the following prompts as appropriate for your network:
9. Respond to the following prompts as appropriate for your network:
10. Respond to the following prompts. Select [2] to save the initial configuration:

DETAILED STEPS

Step 1 From the Cisco IOS-XE CLI, enter the **setup** command in privileged EXEC mode:

Example:

```
Router> enable

Password: <password>

Router# setup

         --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]:
```

You are now in the Setup Configuration Utility.

Depending on your router model, the installed interface modules, and the software image, the prompts in the setup command facility vary. The following steps and the user entries (in bold) are shown only as examples.

Note This setup command facility is also entered automatically if there is no configuration on the router when it is booted into Cisco IOS-XE.

Note If you make a mistake while using the setup command facility, you can exit and run the setup command facility again. Press Ctrl-C, and enter the setup command at the privileged EXEC mode prompt (Router#). For more information on using the setup command facility, see *The Setup Command* chapter in *Cisco IOS Configuration Fundamentals Command Reference*, at the following URL: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/fundamentals/command/cf_command_ref.html

Step 2 To proceed using the setup command facility, enter **yes**.

Example:

```
Continue with configuration dialog? [yes/no]:
At any point you may enter a question mark '?' for help.
```

```
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
```

Step 3 To enter the basic management setup, enter **yes**.

Example:

```
Would you like to enter basic management setup? [yes/no]: yes
```

Step 4 Enter a hostname for the router (this example uses 'myrouter'):

Example:

```
Configuring global parameters:
Enter host name [Router]: myrouter
```

Step 5 Enter an enable secret password. This password is encrypted (for more security) and cannot be seen when viewing the configuration.

Example:

```
The enable secret is a password used to protect access to
privileged EXEC and configuration modes. This password, after
entered, becomes encrypted in the configuration.
Enter enable secret: cisco
```

Step 6 Enter an enable password that is different from the enable secret password. This password is *not* encrypted (and is less secure) and can be seen when viewing the configuration.

Example:

```
The enable password is used when you do not specify an
enable secret password, with some older software versions, and
some boot images.
Enter enable password: cisco123
```

Step 7 Enter the virtual terminal password, which prevents unauthenticated access to the router through ports other than the console port:

Example:

```
The virtual terminal password is used to protect
access to the router over a network interface.
Enter virtual terminal password: cisco
```

Step 8 Respond to the following prompts as appropriate for your network:

Example:

```
Configure SNMP Network Management? [no]: yes
    Community string [public]:
```

A summary of the available interfaces is displayed.

Note The interface summary includes interface numbering, which is dependent on the router model and the installed modules and interface cards.

Example:

```
Current interface summary
```

```
Interface      IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/1/0  10.10.10.12     YES DHCP   up                    up
GigabitEthernet0/2/0    unassigned      YES NVRAM  administratively down down
SSLVPN-VIF0                unassigned      NO  unset  up
Any interface listed with OK? value "NO" does not have a valid configuration
```

Step 9 Respond to the following prompts as appropriate for your network:

Example:

```
Configuring interface GigabitEthernet0/1/0
:
  Configure IP on this interface? [yes]: yes
    IP address for this interface [10.10.10.12
]:
    Subnet mask for this interface [255.0.0.0] : 255.255.255.0
    Class A network is 10.0.0.0, 24 subnet bits; mask is /24
```

The following configuration command script was created:

Example:

```
hostname myrouter
enable secret 5 $1$t/Dj$yAeGKviLLZNOBX0b9eifO0 enable password cisco123 line vty 0 4 password cisco
 snmp-server community public !
no ip routing
!
interface GigabitEthernet0/0/0
shutdown
no ip address
!
interface GigabitEthernet0/1/0
no shutdown
ip address 10.10.10.12 255.255.255.0
!
interface GigabitEthernet0/2/0
shutdown
no ip address
!
end
```

Step 10 Respond to the following prompts. Select [2] to save the initial configuration:

Example:

```
[0] Go to the IOS command prompt without saving this config.
[1] Return back to the setup without saving this config.
[2] Save this configuration to nvram and exit.
Enter your selection [2]: 2
Building configuration...
Use the enabled mode 'configure' command to modify this configuration.
Press RETURN to get started! RETURN
```

The user prompt is displayed:

Example:

```
myrouter>
```

Complete the Configuration

When using the Cisco Setup, and after you have provided all the information requested by the facility, the final configuration appears. To complete your router configuration, follow these steps:

SUMMARY STEPS

1. Choose to save the configuration when the facility prompts you to save the configuration.
2. When the messages stop appearing on your screen, press **Return** to get the Router> prompt.
3. Choose to modify the existing configuration or create another configuration. The Router> prompt indicates that you are now at the command-line interface (CLI) and you have just completed a initial router configuration. Nevertheless, this is *not* a complete configuration. At this point, you have two choices:

DETAILED STEPS

Step 1 Choose to save the configuration when the facility prompts you to save the configuration.

- If you answer 'no', the configuration information you entered is *not* saved, and you return to the router enable prompt (Router#). Enter setup to return to the System Configuration Dialog.
- If you answer 'yes', the configuration is saved, and you are returned to the user EXEC prompt (Router>).

Example:

```
Use this configuration? {yes/no} : yes
Building configuration...
Use the enabled mode 'configure' command to modify this configuration.
Press RETURN to get started!
%LINK-3-UPDOWN: Interface Ethernet0/0, changed state to up
%LINK-3-UPDOWN: Interface Ethernet0/1, changed state to up
%LINK-3-UPDOWN: Interface Serial0/0/0, changed state to up
%LINK-3-UPDOWN: Interface Serial0/0/1, changed state to down
%LINK-3-UPDOWN: Interface Serial0/2, changed state to down
%LINK-3-UPDOWN: Interface Serial1/0, changed state to up
%LINK-3-UPDOWN: Interface Serial1/1, changed state to down
%LINK-3-UPDOWN: Interface Serial1/2, changed state to down
<Additional messages omitted.>
```

Step 2 When the messages stop appearing on your screen, press **Return** to get the Router> prompt.

Step 3 Choose to modify the existing configuration or create another configuration. The Router> prompt indicates that you are now at the command-line interface (CLI) and you have just completed a initial router configuration. Nevertheless, this is *not* a complete configuration. At this point, you have two choices:

- Run the setup command facility again, and create another configuration.

Example:

```
Router> enable
Password: password
Router# setup
```

- Modify the existing configuration or configure additional features by using the CLI:

Example:

```
Router> enable
Password: password
```

```
Router# configure terminal
Router(config)#
```

Use Cisco IOS XE CLI—Manual Configuration

This section describes you how to access the command-line interface (CLI) to perform the initial configuration on the router.

> **Note** To configure the initial router settings by using the Cisco IOS CLI, you must set up a console connection.

If the default configuration file is installed on the router prior to shipping, the system configuration dialog message does not appear, To configure the device, follow these steps:

SUMMARY STEPS

1. Enter the appropriate answer when the following system message appears on the router.
2. Press Return to terminate autoinstall and continue with manual configuration:
3. Press Return to bring up the Router> prompt.
4. Type enable to enter privileged EXEC mode:

DETAILED STEPS

Step 1 Enter the appropriate answer when the following system message appears on the router.

Example:

```
--- System Configuration Dialog ---
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Would you like to enter the initial configuration dialog? [yes/no]: no
```

Step 2 Press Return to terminate autoinstall and continue with manual configuration:

Example:

```
Would you like to terminate autoinstall? [yes]
 Return
```

Several messages are displayed, ending with a line similar to the following:

Example:

```
...
Copyright (c) 1986-2012 by cisco Systems, Inc.
Compiled <date
> <time
> by <person
>
```

Step 3 Press Return to bring up the Router> prompt.

Example:

```
...
flashfs[4]: Initialization complete.
Router>
```

Step 4 Type enable to enter privileged EXEC mode:

Example:

```
Router> enable

Router#
```

Configure Cisco 4000 Series ISR Hostname

The hostname is used in CLI prompts and default configuration filenames. If you do not configure the router hostname, the router uses the factory-assigned default hostname "Router."

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname** *name*
4. Verify that the router prompt displays your new hostname.
5. **end**

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	**hostname** *name* **Example:** `Router(config)# hostname myrouter`	Specifies or modifies the hostname for the network server.
Step 4	Verify that the router prompt displays your new hostname. **Example:**	—

	Command or Action	Purpose
	`myrouter(config)#`	
Step 5	**end** **Example:** `myrouter# end`	(Optional) Returns to privileged EXEC mode.

Configure the Enable and Enable Secret Passwords

To provide an additional layer of security, particularly for passwords that cross the network or are stored on a TFTP server, you can use either the **enable password** command or **enable secret** command. Both commands accomplish the same thing—they allow you to establish an encrypted password that users must enter to access privileged EXEC (enable) mode.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm. Use the **enable password** command only if you boot an older image of the Cisco IOS XE software.

For more information, see the "Configuring Passwords and Privileges" chapter in the Cisco IOS Security Configuration Guide . Also see the Cisco IOS Password Encryption Facts tech note and the Improving Security on Cisco Routers tech note.

Note If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **enable password** *password*
4. **enable secret** *password*
5. **end**
6. **enable**
7. **end**

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:**	Enters global configuration mode.

	Command or Action	Purpose
	`Router# configure terminal`	
Step 3	**enable password** *password* **Example:** `Router(config)# enable password pswd2`	(Optional) Sets a local password to control access to various privilege levels. • We recommend that you perform this step only if you boot an older image of the Cisco IOS-XE software or if you boot older boot ROMs that do not recognize the **enable secret** command.
Step 4	**enable secret** *password* **Example:** `Router(config)# enable secret greentree`	Specifies an additional layer of security over the **enable password** command. • Do not use the same password that you entered in Step 3 .
Step 5	**end** **Example:** `Router(config)# end`	Returns to privileged EXEC mode.
Step 6	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Verify that your new enable or enable secret password works.
Step 7	**end** **Example:** `Router(config)# end`	(Optional) Returns to privileged EXEC mode.

Configure the Console Idle Privileged EXEC Timeout

This section describes how to configure the console line's idle privileged EXEC timeout. By default, the privileged EXEC command interpreter waits 10 minutes to detect user input before timing out.

When you configure the console line, you can also set communication parameters, specify autobaud connections, and configure terminal operating parameters for the terminal that you are using. For more information on configuring the console line, see the *Cisco IOS Configuration Fundamentals and Network Management Configuration Guide* . In particular, see the "Configuring Operating Characteristics for Terminals" and "Troubleshooting and Fault Management" chapters.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **exec-timeout** *minutes* [*seconds*]
5. **end**

6. **show running-config**

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	**line console 0** **Example:** `Router(config)# line console 0`	Configures the console line and starts the line configuration command collection mode.
Step 4	**exec-timeout** *minutes* [*seconds*] **Example:** `Router(config-line)# exec-timeout 0 0`	Sets the idle privileged EXEC timeout, which is the interval that the privileged EXEC command interpreter waits until user input is detected. • The example shows how to specify no timeout. Setting the exec-timeout value to 0 will cause the router to never log out after it is logged in. This could have security implications if you leave the console without manually logging out using the disable command.
Step 5	**end** **Example:** `Router(config)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** `Router(config)# show running-config`	Displays the running configuration file. • Verify that you properly configured the idle privileged EXEC timeout.

Examples

The following example shows how to set the console idle privileged EXEC timeout to 2 minutes 30 seconds:

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

The following example shows how to set the console idle privileged EXEC timeout to 30 seconds:

```
line console
 exec-timeout 0 30
```

Gigabit Ethernet Management Interface Overview

The router provides an Ethernet management port named GigabitEthernet0.

The purpose of this interface is to allow users to perform management tasks on the router. It is an interface that should not and often cannot forward network traffic. It ca, however, be used to access the router through Telnet and SSH to perform management tasks on the router. The interface is most useful before a router begins routing, or in troubleshooting scenarios when other forwarding interfaces are inactive.

Note he following aspects of the management ethernet interface:

- The router has one management ethernet interface named GigabitEthernet0.
- IPv4, IPv6, and ARP are the only routed protocols supported for the interface.
- The interface provides a way to access to the router even if forwarding interfaces are not functional, or the IOS process is down.
- The management ethernet interface is part of its own VRF. See the " Management Ethernet Interface VRF " section in the Software Configuration Guide for Cisco 4000 Series ISRs for more details.

Default Gigabit Ethernet Configuration

By default, a forwarding VRF is configured for the interface with a special group named "Mgmt-intf." This cannot be changed. This isolates the traffic on the management interface away from the forwarding plane. The basic configuration is like other interfaces; however, there are many forwarding features that are not supported on these interfaces. No forwarding features can be configured on the GigabitEthernet0 interface as it is only used for management.

```
For example, the default configuration is as follows:
interface GigabitEthernet0
vrf forwarding Mgmt-intf
ip address 172.18.77.212 255.255.255.0
negotiation auto
```

Gigabit Ethernet Port Numbering

The Gigabit Ethernet Management port is always GigabitEthernet0.

The port can be accessed in configuration mode.

```
Router# config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface gigabitethernet0
Router(config-if)#
```

Configure Gigabit Ethernet Interfaces

This sections shows how to assign an IP address and interface description to an Ethernet interface on your router.

For comprehensive configuration information on Gigabit Ethernet interfaces, see the "Configuring LAN Interfaces" chapter of *Cisco IOS Interface and Hardware Component Configuration Guide* , http://www.cisco.com/en/US/docs/ios/12_2/interface/configuration/guide/icflanin.html

For information on interface numbering, see the software configuration guide for your router.

SUMMARY STEPS

1. **enable**
2. **show ip interface brief**
3. **configure terminal**
4. **interface** {**fastethernet** | **gigabitethernet**} **0**/*port*
5. **description** *string*
6. **ip address** *ip-address mask*
7. **no shutdown**
8. **end**
9. **show ip interface brief**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ip interface brief** **Example:** `Router# show ip interface brief`	Displays a brief status of the interfaces that are configured for IP. • Learn which type of Ethernet interface is on your router.
Step 3	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 4	**interface** {**fastethernet** | **gigabitethernet**} **0**/*port* **Example:** `Router(config)# interface gigabitethernet 0/0/0`	Specifies the Ethernet interface and enters interface configuration mode. **Note** For information on interface numbering, see Slots, Subslots (Bay), Ports, and Interfaces in Cisco 4000 Series ISRs, page 1-38 .
Step 5	**description** *string* **Example:** `Router(config-if)# description GE int to 2nd floor south wing`	(Optional) Adds a description to an interface configuration. The description helps you remember what is attached to this interface. The description can be useful for troubleshooting.
Step 6	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 172.16.74.3 255.255.255.0`	Sets a primary IP address for an interface.

	Command or Action	Purpose
Step 7	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables an interface.
Step 8	**end** **Example:** `Router(config)# end`	Returns to privileged EXEC mode.
Step 9	**show ip interface brief** **Example:** `Router# show ip interface brief`	Displays a brief status of the interfaces that are configured for IP. Verify that the Ethernet interfaces are up and configured correctly.

Configuration Examples

Configuring the GigabitEthernet Interface: Example

```
!
interface GigabitEthernet0/0/0
 description GE int to HR group
 ip address 172.16.3.3 255.255.255.0
 duplex auto
 speed auto
 no shutdown
!
```

Sample Output for the show ip interface brief Command

```
Router#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0/1   unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0/2   unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0/3   unassigned      YES NVRAM  administratively down down
GigabitEthernet0       10.0.0.1        YES manual up                    up
```

Specify a Default Route or Gateway of Last Resort

This section describes how to specify a default route with IP routing enabled. For alternative methods of specifying a default route, see the Configuring a Gateway of Last Resort Using IP Commands Technical Specifications Note.

The Cisco IOS-XE software uses the gateway (router) as a last resort if it does not have a better route for a packet and if the destination is not a connected network. This section describes how to select a network as a default route (a candidate route for computing the gateway of last resort). The way in which routing protocols propagate the default route information varies for each protocol.

Configure IP Routing and IP Protocols

For comprehensive configuration information about IP routing and IP routing protocols, see the Configuring IP Routing Protocol-Independent Feature at cisco.com.

IP Routing

IP routing is automatically enabled in the Cisco ISO- XE software. When IP routing is configured, the system will use a configured or learned route to forward packets, including a configured default route.

Note This task section does not apply when IP routing is disabled. To specify a default route when IP routing is disabled, refer to the Configuring a Gateway of Last Resort Using IP Commands Technical Specifications Note at cisco.com.

Default Routes

A router might not be able to determine the routes to all other networks. To provide complete routing capability, the common practice is to use some routers as smart routers and give the remaining routers default routes to the smart router. (Smart routers have routing table information for the entire internetwork.) These default routes can be passed along dynamically, or can be configured into the individual routers.

Most dynamic interior routing protocols include a mechanism for causing a smart router to generate dynamic default information that is then passed along to other routers.

Default Network

If a router has an interface that is directly connected to the specified default network, the dynamic routing protocols running on the router generates or sources a default route. In the case of RIP, the router will advertise the pseudonetwork 0.0.0.0. In the case of IGRP, the network itself is advertised and flagged as an exterior route.

A router that is generating the default for a network may also need a default of its own. One way a router can generate its own default is to specify a static route to the network 0.0.0.0 through the appropriate device.

Gateway of Last Resort

When default information is being passed along through a dynamic routing protocol, no further configuration is required. The system periodically scans its routing table to choose the optimal default network as its default route. In the case of RIP, there is only one choice, network 0.0.0.0. In the case of IGRP, there might be several networks that can be candidates for the system default. The Cisco IOS-XE software uses both administrative distance and metric information to determine the default route (gateway of last resort). The selected default route appears in the gateway of last resort display of the **show ip route** EXEC command.

If dynamic default information is not being passed to the software, candidates for the default route are specified with the **ip default-network** global configuration command. In this usage, the **ip default-network** command takes an unconnected network as an argument. If this network appears in the routing table from any source (dynamic or static), it is flagged as a candidate default route and is a possible choice for the default route.

If the router has no interface on the default network, but does have a route to it, it considers this network as a candidate default path. The route candidates are examined and based on administrative distance and metric, the best one is chosen. The gateway to the best default path becomes the gateway of last resort.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip routing**
4. **ip route** *dest-prefix mask next-hop-ip-address* [*admin-distance*] [**permanent**]
5. Do one of the following:
   - **ip default-network** *network-number*
   -
   - **ip route** *dest-prefix mask next-hop-ip-address*
6. **end**
7. **show ip route**

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	**ip routing** **Example:** `Router(config)# ip routing`	Enables IP routing.
Step 4	**ip route** *dest-prefix mask next-hop-ip-address* [*admin-distance*] [**permanent**] **Example:** `Router(config)# ip route 192.168.24.0 255.255.255.0 172.28.99.2`	Establishes a static route.
Step 5	Do one of the following: • **ip default-network** *network-number* • • **ip route** *dest-prefix mask next-hop-ip-address* **Example:** `Router(config)# ip default-network 192.168.24.0` **Example:**	Selects a network as a candidate route for computing the gateway of last resort. Creates a static route to network 0.0.0.0 0.0.0.0 for computing the gateway of last resort.

	Command or Action	Purpose
	`Router(config)# ip route 0.0.0.0 0.0.0.0 172.28.99.1`	
Step 6	**end** **Example:** `Router(config)# end`	Returns to privileged EXEC mode.
Step 7	**show ip route** **Example:** `Router# show ip route`	Displays the current routing table information. Verify that the gateway of last resort is set.

Configuration Examples

Specifying a Default Route: Example

```
!
ip route 192.168.24.0 255.255.255.0 172.28.99.2
!
ip default-network 192.168.24.0
!
```

Sample Output for the show ip route Command

```
Router# show ip route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX -   EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2 i - IS-IS, su - IS-IS summary, L1 -
IS-IS level-1, L2 - IS-IS level-2 ia - IS-IS inter area, * - candidate default,
U - per-user static route o - ODR, P - periodic downloaded static route, H - NHRP,
l - LISP a - application route + - replicated route, % - next hop override
Gateway of last resort is not set 10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks C
10.0.0.0/24 is directly connected, Loopback1 L 10.0.0.1/32 is directly connected, Loopback1
 Router#
```

Configure Virtual Terminal Lines for Remote Console Access

Virtual terminal (vty) lines are used to allow remote access to the router. This section shows you how to configure the virtual terminal lines with a password, so that only authorized users can remotely access the router.

By default, the router has five virtual terminal lines. However, you can create additional virtual terminal lines. See the Cisco IOS XE Dial Technologies Configuration Guide at http://www.cisco.com/en/US/docs/ios/dial/configuration/guide/2_xe/dia_2_xe_book.html .

Line passwords and password encryption is described in the C isco IOS XE Security Configuration Guide: Secure Connectivity document available at the following URL: http://www.cisco.com/en/US/docs/ios/ios_xe/sec_secure_connectivity/configuration/guide/2_xe/sec_secure_connectivity_xe_book.html

. See the Security with Passwords, Privilege Levels, and Login Usernames for CLI Sessions on Networking Devices section. If you want to secure the virtual terminal lines (vty) with an access list, see the Access Control Lists: Overview and Guidelines.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty** *line-number* [*ending-line-number*]
4. **password** *password*
5. **login**
6. **end**
7. **show running-config**
8. From another network device, attempt to open a Telnet session to the router.

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	**line vty** *line-number* [*ending-line-number*] **Example:** `Router(config)# line vty 0 4`	Starts the line configuration command collection mode for the virtual terminal lines (vty) for remote console access. • Make sure that you configure all vty lines on your router. **Note** To verify the number of vty lines on your router, use the **line vty ?** command.
Step 4	**password** *password* **Example:** `Router(config-line)# password guessagain`	Specifies a password on a line.
Step 5	**login** **Example:** `Router(config-line)# login`	Enables password checking at login.
Step 6	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	`Router(config-line)# end`	
Step 7	**show running-config** **Example:** `Router# show running-config`	Displays the running configuration file. Verify that you bave properly configured the virtual terminal lines for remote access.

Step 8	From another network device, attempt to open a Telnet session to the router.
	Example:
	`Router# 172.16.74.3`
	Example:
	`Password:`
		Verifies that you can remotely access the router and that the virtual terminal line password is correctly configured.

Configuration Examples

The following example shows how to configure virtual terminal lines with a password:

```
!
line vty 0 4
 password guessagain
 login
!
```

What to Do Next

After you configure the vty lines, follow these steps:

- (Optional) To encrypt the virtual terminal line password, see the "Configuring Passwords and Privileges" chapter in the Cisco IOS Security Configuration Guide . Also see the Cisco IOS Password Encryption Facts tech note.
- (Optional) To secure the VTY lines with an access list, see the "Part 3: Traffic Filtering and Firewalls" in the Cisco IOS Security Configuration Guide .

Configure the Auxiliary Line

This section describes how to enter line configuration mode for the auxiliary line. How you configure the auxiliary line depends on your particular implementation of the auxiliary (AUX) port. See the following documents for information on configuring the auxiliary line:

- *Configuring a Modem on the AUX Port for EXEC Dialin Connectivity* , Technical Specifications Note http://www.cisco.com/en/US/tech/tk801/tk36/technologies_tech_note09186a0080094bbc.shtml
- *Configuring Dialout Using a Modem on the AUX Port* , sample configuration http://www.cisco.com/en/US/tech/tk801/tk36/technologies_configuration_example09186a0080094579.shtml
- *Configuring AUX-to-AUX Port Async Backup with Dialer Watch* , sample configuration http://www.cisco.com/en/US/tech/tk801/tk36/technologies_configuration_example09186a0080093d2b.shtml
- *Modem-Router Connection Guide* , Technical Specifications Note http://www.cisco.com/en/US/tech/tk801/tk36/technologies_tech_note09186a008009428b.shtml

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line aux 0**
4. See the Technical Specifications Note and sample configurations to configure the line for your particular implementation of the AUX port.

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	**line aux 0** **Example:** `Router(config)# line aux 0`	Starts the line configuration command collection mode for the auxiliary line.
Step 4	See the Technical Specifications Note and sample configurations to configure the line for your particular implementation of the AUX port.	—

Verify Network Connectivity

This section describes how to verify network connectivity for your router.

Before you begin

- All configuration tasks describe in this chapter must be completed.
- The router must be connected to a properly configured network host.

SUMMARY STEPS

1. **enable**
2. **ping** [*ip-address* | *hostname*]
3. **telnet** {*ip-address* | *hostname*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**ping** [*ip-address* | *hostname*] **Example:** `Router# ping 172.16.74.5`	Diagnoses initial network connectivity. To verify connectivity, ping the next hop router or connected host for each configured interface to.
Step 3	**telnet** {*ip-address* | *hostname*} **Example:** `Router# telnet 10.20.30.40`	Logs in to a host that supports Telnet. If you want to test the vty line password, perform this step from a different network device, and use your router's IP address.

Examples

The following display shows sample output for the ping command when you ping the IP address 192.168.7.27:

```
Router# ping

Protocol [ip]:
Target IP address: 192.168.7.27

Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.7.27, timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/2/4 ms
```

The following display shows sample output for the ping command when you ping the IP hostname donald:

```
Router# ping donald

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.7.27, timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/3/4 ms
```

Save Your Device Configuration

This section describes how to avoid losing your configuration at the next system reload or power cycle by saving the running configuration to the startup configuration in NVRAM. The NVRAM provides 256KB of storage on the router.

SUMMARY STEPS

1. **enable**
2. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**copy running-config startup-config** **Example:** `Router# copy running-config startup-config`	Saves the running configuration to the startup configuration.

Save Backup Copies of Configuration and System Image

To aid file recovery and minimize downtime in case of file corruption, we recommend that you save backup copies of the startup configuration file and the Cisco IOS-XE software system image file on a server.

SUMMARY STEPS

1. **enable**
2. **copy nvram:startup-config** {**ftp:** | **rcp:** | **tftp:**}
3. **show {bootflash0|bootflash1}:**
4. **copy {bootflash0|bootflash1}:** {**ftp:** | **rcp:** | **tftp:**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**copy nvram:startup-config** {**ftp:** | **rcp:** | **tftp:**} **Example:** `Router# copy nvram:startup-config ftp:`	Copies the startup configuration file to a server. The configuration file copy can serve as a backup copy.Enter the destination URL when prompted.
Step 3	**show {bootflash0|bootflash1}:** **Example:** `Router# show {bootflash0|bootflash1}:`	Displays the layout and contents of a flash memory file system. Learn the name of the system image file.

	Command or Action	Purpose
Step 4	**copy {bootflash0|bootflash1}: {ftp: | rcp: | tftp:}** **Example:** `Router# copy {bootflash0|bootflash1}: ftp:`	Copies a file from flash memory to a server. • Copy the system image file to a server to serve as a backup copy. • Enter the filename and destination URL when prompted.

Configuration Examples

Copying the Startup Configuration to a TFTP Server: Example

The following example shows the startup configuration being copied to a TFTP server:

```
Router# copy nvram:startup-config tftp:

Remote host[]? 172.16.101.101

Name of configuration file to write [rtr2-confg]? <cr>

Write file rtr2-confg on host 172.16.101.101?[confirm] <cr>

![OK]
```

Copying from Flash Memory to a TFTP Server: Example

The following example shows the use of the **show {flash0|flash1}:** command in privileged EXEC to learn the name of the system image file and the use of the **copy {flash0|flash1}: tftp:** privileged EXEC command to copy the system image to a TFTP server. The router uses the default username and password.

```
Router#Directory of bootflash:
11 drwx 16384 Jun 12 2012 17:31:45 +00:00 lost+found 64897 drwx 634880 Sep 6 2012 14:33:26
 +00:00 core 340705 drwx 4096 Oct 11 2012 19:28:27 +00:00 .prst_sync 81121 drwx 4096 Jun
12 2012 17:32:39 +00:00 .rollback_timer 12 -rw- 0 Jun 12 2012 17:32:50 +00:00 tracelogs.336
 713857 drwx 1347584 Oct 11 2012 20:24:26 +00:00 tracelogs 162241 drwx 4096 Jun 12 2012
17:32:51 +00:00 .installer 48673 drwx 4096 Jul 2 2012 17:14:51 +00:00 vman_fdb 13 -rw-
420654048 Aug 28 2012 15:01:31 +00:00
crankshaft-universalk9.BLD_MCP_DEV_LATEST_20120826_083012.SSA.bin 14 -rw- 727035 Aug 29
2012 21:03:25 +00:00 uut2_2000_ikev1.cfg 15 -rw- 420944032 Aug 29 2012 19:40:28 +00:00
crankshaft-universalk9.BLD_MCP_DEV_LATEST_20120829_033026.SSA.bin 16 -rw- 1528 Aug 30 2012
 14:24:38 +00:00 base.cfg 17 -rw- 360900 Aug 31 2012 19:10:02 +00:00 uut2_1000_ikev1.cfg
18 -rw- 421304160 Aug 31 2012 16:34:19 +00:00
crankshaft-universalk9.BLD_MCP_DEV_LATEST_20120821_193221.SSA.bin 19 -rw- 421072064 Aug 31
 2012 18:31:57 +00:00 crankshaft-universalk9.BLD_MCP_DEV_LATEST_20120830_110615.SSA.bin 20
 -rw- 453652 Sep 1 2012 01:48:15 +00:00 uut2_1000_ikev1_v2.cfg 21 -rw- 16452768 Sep 11 2012
 20:36:20 +00:00 upgrade_stage_1_of_1.bin.2012-09-05-Delta 22 -rw- 417375456 Sep 12 2012
20:28:23 +00:00 crankshaft-universalk9.2012-09-12_00.45_cveerapa.SSA.bin 23 -rw- 360879 Oct
 8 2012 19:43:36 +00:00 old-config.conf 24 -rw- 390804800 Oct 11 2012 15:34:08 +00:00
_1010t.bin 7451738112 bytes total (4525948928 bytes free)
Router#show bootflash: -#- --length-- ---------date/time--------- path 1 4096 Oct 11 2012
20:22:19 +00:00 /bootflash/ 2 16384 Jun 12 2012 17:31:45 +00:00 /bootflash/lost+found 3
634880 Sep 06 2012 14:33:26 +00:00 /bootflash/core 4 1028176 Sep 06 2012 14:31:17 +00:00
/bootflash/core/UUT2_RP_0_iomd_17360.core.gz 5 1023738 Sep 06 2012 14:31:24 +00:00
/bootflash/core/UUT2_RP_0_iomd_23385.core.gz 6 1023942 Sep 06 2012 14:31:30 +00:00
/bootflash/core/UUT2_RP_0_iomd_24973.core.gz 7 1023757 Sep 06 2012 14:31:37 +00:00
/bootflash/core/UUT2_RP_0_iomd_26241.core.gz 8 1023726 Sep 06 2012 14:31:43 +00:00
/bootflash/core/UUT2_RP_0_iomd_27507.core.gz 9 1023979 Sep 06 2012 14:31:50 +00:00
```

```
/bootflash/core/UUT2_RP_0_iomd_28774.core.gz 10 1023680 Sep 06 2012 14:31:56 +00:00
/bootflash/core/UUT2_RP_0_iomd_30045.core.gz 11 1023950 Sep 06 2012 14:32:02 +00:00
/bootflash/core/UUT2_RP_0_iomd_31332.core.gz 12 1023722 Sep 06 2012 14:32:09 +00:00
/bootflash/core/UUT2_RP_0_iomd_5528.core.gz 13 1023852 Sep 06 2012 14:32:15 +00:00
/bootflash/core/UUT2_RP_0_iomd_7950.core.gz 14 1023916 Sep 06 2012 14:32:22 +00:00
/bootflash/core/UUT2_RP_0_iomd_9217.core.gz 15 1023875 Sep 06 2012 14:32:28 +00:00
/bootflash/core/UUT2_RP_0_iomd_10484.core.gz 16 1023907 Sep 06 2012 14:32:35 +00:00
/bootflash/core/UUT2_RP_0_iomd_11766.core.gz 17 1023707 Sep 06 2012 14:32:41 +00:00
/bootflash/core/UUT2_RP_0_iomd_13052.core.gz 18 1023963 Sep 06 2012 14:32:48 +00:00
/bootflash/core/UUT2_RP_0_iomd_14351.core.gz 19 1023915 Sep 06 2012 14:32:54 +00:00
/bootflash/core/UUT2_RP_0_iomd_15644.core.gz 20 1023866 Sep 06 2012 14:33:00 +00:00
/bootflash/core/UUT2_RP_0_iomd_17171.core.gz 21 1023518 Sep 06 2012 14:33:07 +00:00
/bootflash/core/UUT2_RP_0_iomd_18454.core.gz 22 1023938 Sep 06 2012 14:33:13 +00:00
/bootflash/core/UUT2_RP_0_iomd_19741.core.gz 23 1024017 Sep 06 2012 14:33:20 +00:00
/bootflash/core/UUT2_RP_0_iomd_21039.core.gz 24 1023701 Sep 06 2012 14:33:26 +00:00
/bootflash/core/UUT2_RP_0_iomd_22323.core.gz 25 4096 Oct 11 2012 19:28:27 +00:00
/bootflash/.prst_sync 26 4096 Jun 12 2012 17:32:39 +00:00 /bootflash/.rollback_timer 27 0
Jun 12 2012 17:32:50 +00:00 /bootflash/tracelogs.336 28 1347584 Oct 11 2012 20:24:26 +00:00
 /bootflash/tracelogs 29 392 Oct 11 2012 20:22:19 +00:00
/bootflash/tracelogs/inst_cleanup_R0-0.log.gz 30 308 Oct 11 2012 18:39:43 +00:00
/bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011183943.gz 31 308 Oct 11 2012 18:49:44
 +00:00 /bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011184944.gz 32 42853 Oct 04
2012 07:35:39 +00:00 /bootflash/tracelogs/hman_R0-0.log.0498.20121004073539.gz 33 307 Oct
11 2012 18:59:45 +00:00 /bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011185945.gz
34 308 Oct 11 2012 19:19:47 +00:00
/bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011191947.gz 35 307 Oct 11 2012 19:37:14
 +00:00 /bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011193714.gz 36 308 Oct 11
2012 19:47:15 +00:00 /bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011194715.gz 37
308 Oct 11 2012 19:57:16 +00:00
/bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011195716.gz 38 308 Oct 11 2012 20:07:17
 +00:00 /bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011200717.gz 39 307 Oct 11
2012 20:12:18 +00:00 /bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011201218.gz 40
306 Oct 11 2012 20:17:18 +00:00
/bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011201718.gz 41 44220 Oct 10 2012
11:47:42 +00:00 /bootflash/tracelogs/hman_R0-0.log.32016.20121010114742.gz 42 64241 Oct 09
 2012 20:47:59 +00:00 /bootflash/tracelogs/fman-fp_F0-0.log.12268.20121009204757.gz 43 177
 Oct 11 2012 19:27:03 +00:00 /bootflash/tracelogs/inst_compmatrix_R0-0.log.gz 44 307 Oct
11 2012 18:24:41 +00:00 /bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011182441.gz
45 309 Oct 11 2012 18:29:42 +00:00
/bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011182942.gz 46 43748 Oct 06 2012
13:49:19 +00:00 /bootflash/tracelogs/hman_R0-0.log.0498.20121006134919.gz 47 309 Oct 11
2012 18:44:43 +00:00 /bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011184443.gz 48
309 Oct 11 2012 19:04:46 +00:00
/bootflash/tracelogs/inst_cleanup_R0-0.log.0000.20121011190446.gz 49 2729 Oct 09 2012
21:21:49 +00:00 /bootflash/tracelogs/IOSRP_R0-0.log.20011.20121009212149 50 116 Oct 08 2012
 21:06:44 +00:00 /bootflash/tracelogs/binos_log_R0-0.log.20013.20121008210644
```

Note To avoid losing work you have completed, be sure to save your configuration occasionally as you proceed. Use the **copy running-config startup-config** command to save the configuration to NVRAM.

Verify Initial Configuration on Cisco 4000 Series ISRs

Enter the following commands at Cisco IOS-XE to verify the initial configuration on the router:

- **show version**—Displays the system hardware version; the installed software version; the names and sources of configuration files; the boot images; and the amount of installed DRAM, NVRAM, and flash memory.

- **show diag**—Lists and displays diagnostic information about the installed controllers, interface processors, and port adapters.
- **show interfaces**— Shows interfaces are operating correctly and that the interfaces and line protocol are in the correct state; either up or down.
- **show ip interface brief—** Displays a summary status of the interfaces configured for IP protocol.
- **show configuration—** Verifies that you have configured the correct hostname and password.
- **show platform—** Displays the software/rommon version, and so on.

When you have completed and verified the initial configuration, specific features and functions are ready to be configured. See the Software Configuration Guide for the Cisco 4400 and Cisco 4300 Series ISRs.

CHAPTER 3

Basic Router Configuration

This section includes information about some basic router configuration, and contains the following sections:

Default Configuration

When you boot up the router, the router looks for a default file name-the PID of the router. For example, the Cisco 4000 Series Integrated Services Routers look for a file named isr 4451.cfg. The Cisco 4000 Series ISR looks for this file before finding the standard files-router-confg or the ciscortr.cfg.

The Cisco 4000 ISR looks for the isr4451.cfg file in the bootflash. If the file is not found in the bootflash, the router then looks for the standard files-router-confg and ciscortr.cfg. If none of the files are found, the router then checks for any inserted USB that may have stored these files in the same particular order.

Note If there is a configuration file with the PID as its name in an inserted USB, but one of the standard files are in bootflash, the system finds the standard file for use.

Use the **show running-config** command to view the initial configuration, as shown in the following example:

```
Router# show running-config
Building configuration...
Current configuration : 977 bytes
!
version 15.3
service timestamps debug datetime msec
service timestamps log datetime msec
no platform punt-keepalive disable-kernel-core
!
```

```
hostname Router
!
boot-start-marker
boot-end-marker
!
!
vrf definition Mgmt-intf
!
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
!
no aaa new-model
!
ipv6 multicast rpf use-bgp
!
!
multilink bundle-name authenticated
!
!
redundancy
mode none
!

interface GigabitEthernet0/0/0
no ip address
negotiation auto
!
interface GigabitEthernet0/0/1
no ip address
negotiation auto
!
interface GigabitEthernet0/0/2
no ip address
negotiation auto
!
interface GigabitEthernet0/0/3
no ip address
negotiation auto
!
interface GigabitEthernet0
vrf forwarding Mgmt-intf
no ip address
negotiation auto
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!

!
control-plane
!
!
line con 0
stopbits 1
line vty 0 4
login
!
!
```

```
end
```

Configuring Global Parameters

To configure the global parameters for your router, follow these steps.

SUMMARY STEPS

1. **configure terminal**
2. **hostname** *name*
3. **enable secret** *password*
4. **no ip domain-lookup**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`Router> enable`
	`Router# configure terminal`
	`Router(config)#`
		Enters global configuration mode when using the console port.
		Use the following to connect to the router with a remote terminal:
		`telnet router-name or address`
		`Login: login-id`
		`Password: *********`
		`Router> enable`

Step 2	**hostname** *name* **Example:** `Router(config)# hostname Router`	Specifies the name for the router.
Step 3	**enable secret** *password* **Example:** `Router(config)# enable secret cr1ny5ho`	Specifies an encrypted password to prevent unauthorized access to the router.
Step 4	**no ip domain-lookup** **Example:** `Router(config)# no ip domain-lookup`	Disables the router from translating unfamiliar words (typos) into IP addresses. For complete information on global parameter commands, see the Cisco IOS Release Configuration Guide documentation set.

Configuring Gigabit Ethernet Interfaces

To manually define onboard Gigabit Ethernet interfaces, follow these steps, beginning from global configuration mode.

SUMMARY STEPS

1. **interface gigabitethernet** *slot/bay/port*
2. **ip address** *ip-address mask*
3. **ipv6 address** *ipv6-address/prefix*
4. **no shutdown**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface gigabitethernet** *slot/bay/port* **Example:** `Router(config)# interface gigabitethernet 0/0/1`	Enters the configuration mode for a Gigabit Ethernet interface on the router.
Step 2	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 192.168.12.2 255.255.255.0`	Sets the IP address and subnet mask for the specified Gigabit Ethernet interface. Use this Step if you are configuring an IPv4 address.
Step 3	**ipv6 address** *ipv6-address/prefix* **Example:** `Router(config-if)# ipv6 address 2001.db8::ffff:1/128`	Sets the IPv6 address and prefix for the specified Gigabit Ethernet interface. Use this step instead of Step 2, if you are configuring an IPv6 address.
Step 4	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the Gigabit Ethernet interface and changes its state from administratively down to administratively up.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits configuration mode for the Gigabit Ethernet interface and returns to privileged EXEC mode.

Configuring a Loopback Interface

Before you begin

The loopback interface acts as a placeholder for the static IP address and provides default routing information.

To configure a loopback interface, follow these steps.

SUMMARY STEPS

1. **interface** *type number*
2. (Option 1) **ip address** *ip-address mask*
3. (Option 2) **ipv6 address** *ipv6-address/prefix*
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface** *type number* **Example:** `Router(config)# interface Loopback 0`	Enters configuration mode on the loopback interface.
Step 2	(Option 1) **ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 10.108.1.1 255.255.255.0`	Sets the IP address and subnet mask on the loopback interface. (If you are configuring an IPv6 address, use the **ipv6 address** *ipv6-address/prefix* command described below.
Step 3	(Option 2) **ipv6 address** *ipv6-address/prefix* **Example:** `Router(config-if)# 2001:db8::ffff:1/128`	Sets the IPv6 address and prefix on the loopback interface.
Step 4	**exit** **Example:** `Router(config-if)# exit`	Exits configuration mode for the loopback interface and returns to global configuration mode.

Example

Verifying Loopback Interface Configuration

The loopback interface in this sample configuration is used to support Network Address Translation (NAT) on the virtual-template interface. This configuration example shows the loopback interface configured on the Gigabit Ethernet interface with an IP address of 192.0.2.0/24, which acts as a static IP address. The loopback interface points back to virtual-template1, which has a negotiated IP address.

```
!
interface loopback 0
ip address 192.0.2.0 255.255.255.0 (static IP address)
ip nat outside
!
interface Virtual-Template1
ip unnumbered loopback0
no ip directed-broadcast
ip nat outside
```

Enter the **show interface loopback** command. You should see an output similar to the following example:

```
Router# show interface loopback 0
Loopback0 is up, line protocol is up
  Hardware is Loopback
  Internet address is 203.0.113.1/24
  MTU 1514 bytes, BW 8000000 Kbit, DLY 5000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation LOOPBACK, loopback not set
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/0, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Alternatively, use the **ping** command to verify the loopback interface, as shown in the following example:

```
Router# ping 192.0.2.0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.0.2.0, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Hardware Limitations for MAC Filters

This section provides the number and distribution of supported virtual MAC addresses on the Cisco 4000 Series ISRs. The virtual MAC address filters are supported on the following interfaces:

- GigabitEthernet Interface MAC Filters
- TenGigabitEthernet Interface MAC Filters

GigabitEthernet Interface MAC Address Filters

The device supports a set of 32 MAC address filters. You can be use these filters across the four GE ports. Each 4 GEport reserves one entry for the primary MAC address (BIA). You can use the remaining 28 MAC filters for features such as Hot Standby Router Protocol (HSRP).

Note Each port can use any number of the available feature filters. A single port can use a maximum of 28 feature filters. If all the 4 GE ports uses the filters equally, then each port can have a maximum of seven filters.

TenGigabitEthernet Interface MAC Address Filters

The device supports a set of 32 MAC address filters. You can be use these filters across the two 10GE ports. Each 10GE port reserves one entry for the primary MAC address (BIA). You can use the remaining 30 MAC filters for features such as HSRP.

Note Each port can use any number of the available feature filters. A single port can use a maximum of 30 feature filters. If both the ports uses the filters equally, then each port can have a maximum of 15 filters.

This limitation applies to port-channel configuration as well. With port-channel configuration, vMACs are reserved per physical interface even when they are bundled in a single port-channel interface. Therefore, the 30 available MAC filters can be attached to a maximum of 15 port-channels.

MAC Filter Distribution

The following tables provide the MAC filter distribution for the Cisco 4000 Series ISRs:

Table 3: Cisco 4461 ISR MAC Filter Distribution

Interface	Total Filters		Primary MAC Address (BIA)		Feature Filters
Gigabit0/0/0	32	=	1	+	28
Gigabit0/0/1			1		
Gigabit0/0/2			1		
Gigabit0/0/3			1		
TenGigabit0/0/0	32	=	1	+	30
TenGigabit0/0/1			1		

Table 4: Cisco 4451 and 4431 ISRs GigabitEthernet Interface MAC Filters Distribution

Interface	Total Filters		Primary MAC Address (BIA)		Feature Filters
Gigabit0/0/0	32	=	1	+	28
Gigabit0/0/1			1		
Gigabit0/0/2			1		
Gigabit0/0/3			1		

Table 5: Cisco ISR4351 and 4331 ISR MAC Filter Distribution

Interface	Total Filters		Primary MAC Address (BIA)		Feature Filters
Gigabit0/0/0	16	=	1	+	15
Gigabit0/0/1	16		1		15
Gigabit0/0/2	16		1		15

Table 6: Cisco 4321 and 4221 ISRs MAC Filter Distribution

Interface	Total Filters		Primary MAC Address (BIA)		Feature Filters
Gigabit0/0/0	16	=	1	+	15
Gigabit0/0/1	16	=	1	+	15

Configuring Module Interfaces

For detailed information about configuring service modules, see "Service Modules" in the "Service Module Management" section of the Cisco SM-1T3/E3 Service Module Configuration Guide.

Enabling Cisco Discovery Protocol

Cisco Discovery Protocol (CDP) is enabled by default on the router.

Note CDP is not enabled by default on Cisco Aggregation Services Routers or on the Cisco CSR 1000v.

For more information on using CDP, see Cisco Discovery Protocol Configuration Guide, Cisco IOS XE Release 3S.

Configuring Command-Line Access

To configure parameters to control access to the router, follow these steps.

SUMMARY STEPS

1. **line** [**aux** | **console** | **tty** | **vty**] *line-number*
2. **password** *password*
3. **login**
4. **exec-timeout** *minutes* [*seconds*]
5. **exit**

6. **line** [**aux** | **console** | **tty** | **vty**] *line-number*
7. **password** *password*
8. **login**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**line** [**aux** | **console** | **tty** | **vty**] *line-number* **Example:** `Router(config)# line console 0`	Enters line configuration mode, and specifies the type of line. The example provided here specifies a console terminal for access.
Step 2	**password** *password* **Example:** `Router(config-line)# password 5dr4Hepw3`	Specifies a unique password for the console terminal line.
Step 3	**login** **Example:** `Router(config-line)# login`	Enables password checking at terminal session login.

Step 4	**exec-timeout** *minutes* [*seconds*]
	Example:
	`Router(config-line)# exec-timeout 5 30`
	`Router(config-line)#`
		Sets the interval during which the EXEC command interpreter waits until user input is detected. The default is 10 minutes. Optionally, adds seconds to the interval value.
		The example provided here shows a timeout of 5 minutes and 30 seconds. Entering a timeout of **0 0** specifies never to time out.

Step 5	**exit** **Example:** `Router(config-line)# exit`	Exits line configuration mode to re-enter global configuration mode.

Step 6	**line** [**aux** | **console** | **tty** | **vty**] *line-number*
	Example:
	`Router(config)# line vty 0 4`
	`Router(config-line)#`
		Specifies a virtual terminal for remote console access.

Step 7	**password** *password* **Example:** `Router(config-line)# password aldf2ad1`	Specifies a unique password for the virtual terminal line.
Step 8	**login** **Example:**	Enables password checking at the virtual terminal session login.

	Command or Action	Purpose
	`Router(config-line)# login`	
Step 9	**end** **Example:** `Router(config-line)# end`	Exits line configuration mode, and returns to privileged EXEC mode.

Example

The following configuration shows the command-line access commands.

You do not have to input the commands marked **default**. These commands appear automatically in the configuration file that is generated when you use the **show running-config** command.

```
!
line console 0
exec-timeout 10 0
password 4youreyesonly
login
transport input none (default)
stopbits 1 (default)
line vty 0 4
password secret
login
!
```

Configuring Static Routes

Static routes provide fixed routing paths through the network. They are manually configured on the router. If the network topology changes, the static route must be updated with a new route. Static routes are private routes unless they are redistributed by a routing protocol.

To configure static routes, follow these steps.

SUMMARY STEPS

1. (Option 1) **ip route** *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]}
2. (Option 2) **ipv6 route** *prefix/mask* {*ipv6-address* | *interface-type interface-number* [*ipv6-address*]}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Option 1) **ip route** *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} **Example:**	Specifies a static route for the IP packets. (If you are configuring an IPv6 address, use the **ipv6 route** command described below.)

	Command or Action	Purpose
	`Router(config)# ip route 192.168.1.0 255.255.0.0 10.10.10.2`	
Step 2	(Option 2) **ipv6 route** *prefix/mask* {*ipv6-address* | *interface-type interface-number* [*ipv6-address*]} **Example:** `Router(config)# ipv6 route 2001:db8:2::/64`	Specifies a static route for the IP packets.
Step 3	**end** **Example:** `Router(config)# end`	Exits global configuration mode and enters privileged EXEC mode.

Example

Verifying Configuration

In the following configuration example, the static route sends out all IP packets with a destination IP address of 192.168.1.0 and a subnet mask of 255.255.255.0 on the Gigabit Ethernet interface to another device with an IP address of 10.10.10.2. Specifically, the packets are sent to the configured PVC.

You do not have to enter the command marked **default**. This command appears automatically in the configuration file generated when you use the **running-config** command.

```
!
ip classless (default)
ip route 192.168.1.0 255.255.255.0
```

To verify that you have configured static routing correctly, enter the **show ip route** command (or **show ipv6 route** command) and look for static routes marked with the letter S.

When you use an IPv4 address, you should see verification output similar to the following:

```
Router# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.10.10.2/24 is subnetted, 1 subnets
C       10.10.10.2 is directly connected, Loopback0
S*   0.0.0.0/0 is directly connected, FastEthernet0
```

When you use an IPv6 address, you should see verification output similar to the following:

```
Router# show ipv6 route
```

```
IPv6 Routing Table - default - 5 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE -
Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       ls - LISP site, ld - LISP dyn-EID, a - Application

C   2001:DB8:3::/64 [0/0]
       via GigabitEthernet0/0/2, directly connected
S   2001:DB8:2::/64 [1/0]
       via 2001:DB8:3::1
```

Configuring Dynamic Routes

In dynamic routing, the network protocol adjusts the path automatically, based on network traffic or topology. Changes in dynamic routes are shared with other routers in the network.

A router can use IP routing protocols, such as Routing Information Protocol (RIP) or Enhanced Interior Gateway Routing Protocol (EIGRP), to learn about routes dynamically.

- Configuring Routing Information Protocol, on page 40
- Configuring Enhanced Interior Gateway Routing Protocol, on page 43

Configuring Routing Information Protocol

To configure the RIP on a router, follow these steps.

SUMMARY STEPS

1. **router rip**
2. **version** {**1** | **2**}
3. **network** *ip-address*
4. **no auto-summary**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**router rip** **Example:** `Router(config)# router rip`	Enters router configuration mode, and enables RIP on the router.
Step 2	**version** {**1** | **2**} **Example:** `Router(config-router)# version 2`	Specifies use of RIP version 1 or 2.

	Command or Action	Purpose

Step 3	**network** *ip-address*
	Example:
	`Router(config-router)# network 192.168.1.1`
	`Router(config-router)# network 10.10.7.1`
		Specifies a list of networks on which RIP is to be applied, using the address of the network of each directly connected network.

Step 4	**no auto-summary** **Example:** `Router(config-router)# no auto-summary`	Disables automatic summarization of subnet routes into network-level routes. This allows subprefix routing information to pass across classful network boundaries.
Step 5	**end** **Example:** `Router(config-router)# end`	Exits router configuration mode, and enters privileged EXEC mode.

Example

Verifying Configuration

The following configuration example shows RIP Version 2 enabled in IP networks 10.0.0.0 and 192.168.1.0. To see this configuration, use the **show running-config** command from privileged EXEC mode.

```
!
Router# show running-config
Building configuration...

Current configuration : 1616 bytes
!
! Last configuration change at 03:17:14 EST Thu Sep 6 2012
!
version 15.3
service timestamps debug datetime msec
service timestamps log datetime msec
no platform punt-keepalive disable-kernel-core
!
hostname Router
!
boot-start-marker
boot-end-marker
!
!
vrf definition Mgmt-intf
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
!
enable password cisco
!
```

```
no aaa new-model
!
transport-map type console consolehandler
 banner wait ^C
Waiting for IOS vty line
^C
 banner diagnostic ^C
Welcome to diag mode
^C
!
clock timezone EST -4 0
!
!


ip domain name cisco.com
ip name-server vrf Mgmt-intf 203.0.113.1
ip name-server vrf Mgmt-intf 203.0.113.129

!
ipv6 multicast rpf use-bgp
!
!
multilink bundle-name authenticated
!
redundancy
 mode none
!
ip ftp source-interface GigabitEthernet0
ip tftp source-interface GigabitEthernet0
!
!
interface GigabitEthernet0/0/0
 no ip address
 negotiation auto
!
interface GigabitEthernet0/0/1
 no ip address
 negotiation auto
!
interface GigabitEthernet0/0/2
 no ip address
 negotiation auto
!
interface GigabitEthernet0/0/3
 no ip address
 negotiation auto
!
interface GigabitEthernet0
 vrf forwarding Mgmt-intf
 ip address 172.18.77.212 255.255.255.240
 negotiation auto
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
ip route vrf Mgmt-intf 0.0.0.0 0.0.0.0 172.18.77.209
!
control-plane
!
!
line con 0
 stopbits 1
```

```
line aux 0
 stopbits 1
line vty 0 4
 password cisco
 login
!
transport type console 0 input consolehandler
!
ntp server vrf Mgmt-intf 10.81.254.131
!
end
```

To verify that you have configured RIP correctly, enter the **show ip route** command and look for RIP routes marked with the letter R. You should see an output similar to the one shown in the following example:

```
Router# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 1 subnets
C       10.108.1.0 is directly connected, Loopback0
R    10.0.0.0/8 [120/1] via 10.2.2.1, 00:00:02, Ethernet0/0/0
```

Configuring Enhanced Interior Gateway Routing Protocol

To configure Enhanced Interior Gateway Routing Protocol (EIGRP), follow these steps.

SUMMARY STEPS

1. **router eigrp** *as-number*
2. **network** *ip-address*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**router eigrp** *as-number* **Example:** `Router(config)# router eigrp 109`	Enters router configuration mode, and enables EIGRP on the router. The autonomous-system number identifies the route to other EIGRP routers and is used to tag the EIGRP information.

Step 2	**network** *ip-address*
	Example:
	`Router(config)# network 192.168.1.0`
	`Router(config)# network 10.10.12.115`
		Specifies a list of networks on which EIGRP is to be applied, using the IP address of the network of directly connected networks.

	Command or Action	Purpose
Step 3	**end** **Example:** `Router(config-router)# end`	Exits router configuration mode, and enters privileged EXEC mode.

Example

Verifying the Configuration

The following configuration example shows the EIGRP routing protocol enabled in IP networks 192.168.1.0 and 10.10.12.115. The EIGRP autonomous system number is 109. To see this configuration, use the **show running-config** command.

```
Router# show running-config
.
.
.
!
router eigrp 109
 network 192.168.1.0
  network 10.10.12.115
!
.
.
.
```

To verify that you have configured IP EIGRP correctly, enter the **show ip route** command, and look for EIGRP routes marked by the letter D. You should see verification output similar to the following:

```
Router# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 1 subnets
C       10.108.1.0 is directly connected, Loopback0
D     10.0.0.0/8 [90/409600] via 10.2.2.1, 00:00:02, Ethernet0/0
```

CHAPTER 4

Using Cisco IOS XE Software

This chapter describes the basics of using the Cisco IOS XE software and includes the following section:

Accessing the CLI Using a Router Console

Before you begin

There are two serial ports: a console (CON) port and an auxiliary (AUX) port. Use the CON port to access the command-line interface (CLI) directly or when using Telnet.

The following sections describe the main methods of accessing the router:

Accessing the CLI Using a Directly-Connected Console

The CON port is an EIA/TIA-232 asynchronous, serial connection with no-flow control and an RJ-45 connector. The CON port is located on the front panel of the chassis.

The following sections describe the procedure to access the control interface:

Connecting to the Console Port

Step 1 Configure your terminal emulation software with the following settings:

- 9600 bits per second (bps)

- 8 data bits
- No parity
- No flow control

Step 2 Connect to the CON port using the RJ-45-to-RJ-45 cable and the RJ-45-to-DB-25 DTE adapter or the RJ-45-to-DB-9 DTE adapter (labeled Terminal).

Using the Console Interface

Step 1 Enter the following command:

```
Router> enable
```

Step 2 (Go to Step 3 if the enable password has not been configured.) At the password prompt, enter your system password:

```
Password: enablepass
```

When your password is accepted, the privileged EXEC mode prompt is displayed.

```
Router#
```

You now have access to the CLI in privileged EXEC mode and you can enter the necessary commands to complete your desired tasks.

Step 3 If you enter the **setup** command, see “Using Cisco Setup Command Facility” in the “Initial Configuration” section of the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

Step 4 To exit the console session, enter the **quit** command:

```
Router# quit
```

Using SSH to Access Console

Secure Shell (SSH) is a protocol which provides a secure remote access connection to network devices. To enable SSH support on the device:

Step 1 Configure the hostname:

```
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname xxx_lab
```

Here, *host name* is the router hostname or IP address.

Step 2 Configure the DNS domain of the router:

```
xxx_lab(config)# xxx.cisco.com
```

Step 3 Generate an SSH key to be used with SSH:

```
xxx_lab(config)#  crypto key generate rsa
The name for the keys will be: xxx_lab.xxx.cisco.com Choose the size of the key modulus in the range
```

```
of 360 to 4096 for your General Purpose Keys. Choosing a key modulus greater than 512 may take a few
minutes.
How many bits in the modulus [512]: 1024 % Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 0 seconds)
xxx_lab(config)#
```

Step 4 By default, the vtys? transport is Telnet. In this case, Telnet is disabled and only SSH is supported:

```
xxx_lab(config)#line vty 0 4
xxx_lab(config-line)#transport input SSH
```

Step 5 Create a username for SSH authentication and enable login authentication:

```
xxx_lab(config)# username jsmith privilege 15 secret 0 p@ss3456
xxx_lab(config)#line vty 0 4
xxx_lab(config-line)# login local
```

Step 6 Verify remote connection to the device using SSH.

Accessing the CLI from a Remote Console Using Telnet

The following topics describe the procedure to access the CLI from a remote console using Telnet:

• Preparing to Connect to the Router Console Using Telnet, on page 47
• Using Telnet to Access a Console Interface, on page 48

Preparing to Connect to the Router Console Using Telnet

To access the router remotely using Telnet from a TCP/IP network, configure the router to support virtual terminal lines using the **line vty** global configuration command. Configure the virtual terminal lines to require users to log in and specify a password.

See the Cisco IOS Terminal Services Command Reference document for more information about the line **vty global** configuration command.

To prevent disabling login on a line, specify a password with the **password** command when you configure the **login** command.

If you are using authentication, authorization, and accounting (AAA), configure the **login authentication** command. To prevent disabling login on a line for AAA authentication when you configure a list with the login authentication command, you must also configure that list using the **aaa authentication login** global configuration command.

For more information about AAA services, see the Cisco IOS XE Security Configuration Guide: Secure Connectivity and the Cisco IOS Security Command Reference documents. For more information about the **login line-configuration** command, see the Cisco IOS Terminal Services Command Reference document.

In addition, before you make a Telnet connection to the router, you must have a valid hostname for the router or have an IP address configured on the router. For more information about the requirements for connecting to the router using Telnet, information about customizing your Telnet services, and using Telnet key sequences, see the Cisco IOS Configuration Fundamentals Configuration Guide.

Using Telnet to Access a Console Interface

Step 1 From your terminal or PC, enter one of the following commands:

- **connect host** [*port*] [*keyword*]
- **telnet host** [*port*] [*keyword*]

Here, *host* is the router hostname or IP address, *port* is a decimal port number (23 is the default), and *keyword* is a supported keyword. For more information about these commands, see the Cisco IOS Terminal Services Command Reference document.

Note If you are using an access server, specify a valid port number, such as **telnet 172.20.52.40 2004**, in addition to the hostname or IP address.

The following example shows how to use the **telnet** command to connect to a router named **router**:

```
unix_host% telnet router
Trying 172.20.52.40...
Connected to 172.20.52.40.
Escape character is '^]'.
unix_host% connect
```

Step 2 Enter your login password:

```
User Access Verification
Password: mypassword
```

Note If no password has been configured, press **Return**.

Step 3 From user EXEC mode, enter the **enable** command:

```
Router> enable
```

Step 4 At the password prompt, enter your system password:

```
Password: enablepass
```

Step 5 When the **enable** password is accepted, the privileged EXEC mode prompt is displayed:

```
Router#
```

Step 6 You now have access to the CLI in privileged EXEC mode and you can enter the necessary commands to complete your desired tasks.

Step 7 To exit the Telnet session, use the **exit** or **logout** command.

```
Router# logout
```

Accessing the CLI from a USB Serial Console Port

The router provides an additional mechanism for configuring the system: a type B miniport USB serial console that supports remote administration of the router using a type B USB-compliant cable. See the “Connecting to a Console Terminal or Modem” section in the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

Using Keyboard Shortcuts

Commands are not case sensitive. You can abbreviate commands and parameters if the abbreviations contain enough letters to be different from any other currently available commands or parameters.

The following table lists the keyboard shortcuts for entering and editing commands.

Table 7: Keyboard Shortcuts

Key Name	Purpose
Ctrl-B or the **Left Arrow** key[1]	Move the cursor back one character.
Ctrl-F or the **Right Arrow** key[1]	Move the cursor forward one character.
Ctrl-A	Move the cursor to the beginning of the command line.
Ctrl-E	Move the cursor to the end of the command line.
Esc B	Move the cursor back one word.
Esc F	Move the cursor forward one word.

Using the History Buffer to Recall Commands

The history buffer stores the last 20 commands you entered. History substitution allows you to access these commands without retyping them, by using special abbreviated commands.

The following table lists the history substitution commands.

Table 8: History Substitution Commands

Command	Purpose
Ctrl-P or the **Up Arrow** key[1]	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or the **Down Arrow** key[1]	Returns to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the **Up Arrow** key.
Router# show history	While in EXEC mode, lists the last few commands you entered.

[1] The arrow keys function only on ANSI-compatible terminals such as VT100s.

Understanding Command Modes

The command modes available in Cisco IOS XE are the same as those available in traditional Cisco IOS. Use the CLI to access Cisco IOS XE software. Because the CLI is divided into many different modes, the commands available to you at any given time depend on the mode that you are currently in. Entering a question mark (**?**) at the CLI prompt allows you to obtain a list of commands available for each command mode.

When you log in to the CLI, you are in user EXEC mode. User EXEC mode contains only a limited subset of commands. To have access to all commands, you must enter privileged EXEC mode, normally by using a password. From privileged EXEC mode, you can issue any EXEC command—user or privileged mode—or you can enter global configuration mode. Most EXEC commands are one-time commands. For example, **show** commands show important status information, and **clear** commands clear counters or interfaces. The EXEC commands are not saved when the software reboots.

Configuration modes allow you to make changes to the running configuration. If you later save the running configuration to the startup configuration, these changed commands are stored when the software is rebooted. To enter specific configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and a variety of other modes, such as protocol-specific modes.

ROM monitor mode is a separate mode used when the Cisco IOS XE software cannot load properly. If a valid software image is not found when the software boots or if the configuration file is corrupted at startup, the software might enter ROM monitor mode.

The following table describes how to access and exit various common command modes of the Cisco IOS XE software. It also shows examples of the prompts displayed for each mode.

Table 9: Accessing and Exiting Command Modes

Command Mode	Access Method	Prompt	Exit Method
User EXEC	Log in.	`Router>`	Use the **logout** command.
Privileged EXEC	From user EXEC mode, use the **enable** command.	`Router#`	To return to user EXEC mode, use the **disable** command.
Global configuration	From privileged EXEC mode, use the **configure terminal** command.	`Router(config)#`	To return to privileged EXEC mode from global configuration mode, use the **exit** or **end** command.
Interface configuration	From global configuration mode, specify an interface using an **interface** command.	`Router(config-if)#`	To return to global configuration mode, use the **exit** command. To return to privileged EXEC mode, use the **end** command.

Command Mode	Access Method	Prompt	Exit Method
Diagnostic	The router boots up or accesses diagnostic mode in the following scenarios: • In some cases, diagnostic mode will be reached when the Cisco IOS process or processes fail. In most scenarios, however, the router will reload. • A user-configured access policy is configured using the **transport-map** command that directs a user into diagnostic mode. • A break signal (**Ctrl-C**, **Ctrl-Shift-6**, or the **send break** command) is entered and the router is configured to go to diagnostic mode when the break signal is received.	`Router(diag)#`	If failure of the Cisco IOS process is the reason for entering diagnostic mode, the Cisco IOS problem must be resolved and the router rebooted to get out of diagnostic mode. If the router is in diagnostic mode because of a transport-map configuration, access the router through another port or by using a method that is configured to connect to the Cisco IOS CLI.
ROM monitor	From privileged EXEC mode, use the **reload** EXEC command. Press the **Break** key during the first 60 seconds while the system is booting.	`rommon#>`	To exit ROM monitor mode, manually boot a valid image or perform a reset with autoboot set so that a valid image is loaded.

Understanding Diagnostic Mode

The router boots up or accesses diagnostic mode in the following scenarios:

- The IOS process or processes fail, in some scenarios. In other scenarios, the system resets when the IOS process or processes fail.
- A user-configured access policy was configured using the **transport-map** command that directs the user into the diagnostic mode.
- A send break signal (**Ctrl-C** or **Ctrl-Shift-6**) was entered while accessing the router, and the router was configured to enter diagnostic mode when a break signal was sent.

In the diagnostic mode, a subset of the commands that are available in user EXEC mode are made available to the users. Among other things, these commands can be used to:

- Inspect various states on the router, including the IOS state.
- Replace or roll back the configuration.
- Provide methods of restarting the IOS or other processes.
- Reboot hardware, such as the entire router, a module, or possibly other hardware components.
- Transfer files into or off of the router using remote access methods such as FTP, TFTP, and SCP.

The diagnostic mode provides a more comprehensive user interface for troubleshooting than previous routers, which relied on limited access methods during failures, such as ROMMON, to diagnose and troubleshoot Cisco IOS problems. The diagnostic mode commands can work when the Cisco IOS process is not working properly. These commands are also available in privileged EXEC mode on the router when the router is working normally.

Getting Help

Entering a question mark (**?**) at the CLI prompt displays a list of commands available for each command mode. You can also get a list of keywords and arguments associated with any command by using the context-sensitive help feature.

To get help that is specific to a command mode, a command, a keyword, or an argument, use one of the following commands.

Command	Purpose
help	Provides a brief description of the help system in any command mode.
abbreviated-command-entry?	Provides a list of commands that begin with a particular character string. **Note** There is no space between the command and the question mark.
abbreviated-command-entry<Tab>	Completes a partial command name.
?	Lists all the commands that are available for a particular command mode.
command ?	Lists the keywords or arguments that you must enter next on the command line. **Note** There is a space between the command and the question mark.

Finding Command Options: Example

This section provides information about how to display the syntax for a command. The syntax can consist of optional or required keywords and arguments. To display keywords and arguments for a command, enter a question mark (**?**) at the configuration prompt or after entering a part of a command followed by a space. The

Cisco IOS XE software displays a list and brief descriptions of the available keywords and arguments. For example, if you are in global configuration mode and want to see all the keywords and arguments for the **arap** command, you should type **arap ?**.

The <cr> symbol in command help output stands for carriage return. On older keyboards, the carriage return key is the **Return** key. On most modern keyboards, the carriage return key is the **Enter** key. The <cr> symbol at the end of command help output indicates that you have the option to press **Enter** to complete the command and that the arguments and keywords in the list preceding the <cr> symbol are optional. The <cr> symbol by itself indicates that no more arguments or keywords are available, and that you must press **Enter** to complete the command.

The following table shows examples of using the question mark (**?**) to assist you in entering commands.

Table 10: Finding Command Options

Command	Comment
Router> **enable** Password: <password> Router#	Enter the **enable** command and password to access privileged EXEC commands. You are in privileged EXEC mode when the prompt changes to a “ # ” from the “ > ”, for example, Router> to Router#
Router# **configure terminal** Enter configuration commands, one per line. End with CNTL/Z. Router(config)#	Enter the **configure terminal** privileged EXEC command to enter global configuration mode. You are in global configuration mode when the prompt changes to Router (config)#
Router(config)# **interface GigabitEthernet ?** <0-0> GigabitEthernet interface number <0-2> GigabitEthernet interface number Router(config)# **interface GigabitEthernet 1/?** <0-4> Port Adapter number Router (config)# **interface GigabitEthernet 1/3/?** <0-15> GigabitEthernet interface number Router (config)# **interface GigabitEthernet 1/3/8?** . <0-3> Router (config)# **interface GigabitEthernet 1/3/8.0** Router(config-if)#	Enter interface configuration mode by specifying the interface that you want to configure, using the **interface GigabitEthernet** global configuration command. Enter **?** to display what you must enter next on the command line. When the <cr> symbol is displayed, you can press **Enter** to complete the command. You are in interface configuration mode when the prompt changes to Router(config-if)#

Command	Comment

`Router(config-if)# ?`
`Interface configuration commands:`
`  .`
`  .`
`  .`
`  ip                   Interface Internet`
`Protocol`
`                       config commands`
`  keepalive            Enable keepalive`
`  lan-name             LAN Name command`
`  llc2               LLC2 Interface Subcommands`
 	
`  load-interval       Specify interval for load`
` calculation`
`                       for an interface`
`  locaddr-priority     Assign a priority group`
`  logging              Configure logging for`
`interface`
`  loopback             Configure internal`
`loopback on an`
`                       interface`
`  mac-address          Manually set interface`
`MAC address`
`  mls                  mls router sub/interface`
` commands`
`  mpoa                 MPOA interface`
`configuration commands`
`  mtu                  Set the interface`
`                      Maximum Transmission Unit`
` (MTU)`
`  netbios              Use a defined NETBIOS`
`access list`
`                       or enable`
`                       name-caching`
`  no                   Negate a command or set`
`its defaults`
`  nrzi-encoding        Enable use of NRZI`
`encoding`
`  ntp                  Configure NTP`
`  .`
`  .`
`  .`
`Router(config-if)#`
	Enter **?** to display a list of all the interface configuration commands available for the interface. This example shows only some of the available interface configuration commands.

<table>
<tr><th>Command</th><th>Comment</th></tr>
<tr><td><pre>
Router(config-if)# ip ?
Interface IP configuration subcommands:
  access-group   Specify access control
for packets
  accounting   Enable IP accounting on
this interface
  address   Set the IP address of an
 interface
  authentication   authentication subcommands

  bandwidth-percent   Set EIGRP bandwidth limit

  broadcast-address   Set the broadcast address
 of an interface
  cgmp   Enable/disable CGMP
  directed-broadcast Enable forwarding of
directed broadcasts
  dvmrp   DVMRP interface commands
  hello-interval   Configures IP-EIGRP hello
 interval
  helper-address   Specify a destination
address for UDP broadcasts
  hold-time   Configures IP-EIGRP hold
 time
  .
  .
  .
Router(config-if)# ip
</pre></td><td>Enter the command that you want to configure for the interface. This example uses the ip command.

Enter ? to display what you must enter next on the command line. This example shows only some of the available interface IP configuration commands.</td></tr>
<tr><td><pre>
Router(config-if)# ip address ?
  A.B.C.D   IP address
  negotiated   IP Address negotiated over
 PPP
Router(config-if)# ip address
</pre></td><td>Enter the command that you want to configure for the interface. This example uses the ip address command.

Enter ? to display what you must enter next on the command line. In this example, you must enter an IP address or the negotiated keyword.

A carriage return (<cr>) is not displayed. Therefore, you must enter additional keywords or arguments to complete the command.</td></tr>
<tr><td><pre>
Router(config-if)# ip address 172.16.0.1 ?
  A.B.C.D   IP subnet mask
Router(config-if)# ip address 172.16.0.1
</pre></td><td>Enter the keyword or argument that you want to use. This example uses the 172.16.0.1 IP address.

Enter ? to display what you must enter next on the command line. In this example, you must enter an IP subnet mask.

<cr> is not displayed. Therefore, you must enter additional keywords or arguments to complete the command.</td></tr>
</table>

Command	Comment

`Router(config-if)# ip address 172.16.0.1 255.255.255.0 ?`
`  secondary          Make this IP address a secondary address`
`  <cr>`
`Router(config-if)# ip address 172.16.0.1 255.255.255.0`
	Enter the IP subnet mask. This example uses the 255.255.255.0 IP subnet mask.
	Enter **?** to display what you must enter next on the command line. In this example, you can enter the **secondary** keyword, or you can press **Enter**.
	<cr> is displayed. Press **Enter** to complete the command, or enter another keyword.

`Router(config-if)# ip address 172.16.0.1 255.255.255.0` `Router(config-if)#`	Press **Enter** to complete the command.

Using the no and default Forms of Commands

Almost every configuration command has a **no** form. In general, use the **no** form to disable a function. Use the command without the **no** keyword to re-enable a disabled function or to enable a function that is disabled by default. For example, IP routing is enabled by default. To disable IP routing, use the **no ip routing** command; to re-enable IP routing, use the **ip routing** command. The Cisco IOS software command reference publications provide the complete syntax for the configuration commands and describe what the **no** form of a command does.

Many CLI commands also have a **default** form. By issuing the *<command>* **default** command-name, you can configure the command to its default setting. The Cisco IOS software command reference publications describe the function from a **default** form of the command when the **default** form performs a different function than the plain and **no** forms of the command. To see what default commands are available on your system, enter **default ?** in the appropriate command mode.

Saving Configuration Changes

Use the **copy running-config startup-config** command to save your configuration changes to the startup configuration so that the changes will not be lost if the software reloads or a power outage occurs. For example:

```
Router# copy running-config startup-config
Building configuration...
```

It may take a few minutes to save the configuration. After the configuration has been saved, the following output is displayed:

```
[OK]
Router#
```

This task saves the configuration to the NVRAM.

Managing Configuration Files

The startup configuration file is stored in the nvram: file system and the running configuration files are stored in the system: file system. This configuration file storage setup is also used on several other Cisco router platforms.

As a matter of routine maintenance on any Cisco router, users should back up the startup configuration file by copying the startup configuration file from NVRAM to one of the router's other file systems and, additionally, to a network server. Backing up the startup configuration file provides an easy method of recovering the startup configuration file if the startup configuration file in NVRAM becomes unusable for any reason.

The **copy** command can be used to back up startup configuration files.

For more detailed information on managing configuration files, see the "Managing Configuration Files" section in the Cisco IOS XE Configuration Fundamentals Configuration Guide.

Filtering Output from the show and more Commands

You can search and filter the output of **show** and **more** commands. This functionality is useful if you need to sort through large amounts of output or if you want to exclude output that you need not see.

To use this functionality, enter a **show** or **more** command followed by the "pipe" character (|); one of the keywords **begin**, **include**, or **exclude**; and a regular expression on which you want to search or filter (the expression is case sensitive):

show *command* | {**append** | **begin** | **exclude** | **include** | **redirect** | **section** | **tee**} *regular-expression*

The output matches certain lines of information in the configuration file.

Example

In this example, a modifier of the **show interface** command (**include protocol**) is used to provide only the output lines in which the expression **protocol** is displayed:

```
Router# show interface | include protocol
GigabitEthernet0/0/0 is administratively down, line protocol is down
     0 unknown protocol drops
GigabitEthernet0/0/1 is administratively down, line protocol is down
     0 unknown protocol drops
GigabitEthernet0/0/2 is administratively down, line protocol is down
     0 unknown protocol drops
GigabitEthernet0/0/3 is administratively down, line protocol is down
     0 unknown protocol drops
GigabitEthernet0 is up, line protocol is up
     0 unknown protocol drops
Loopback0 is up, line protocol is up
     0 unknown protocol drops
```

Powering Off a Router

The router can be safely turned off at any time by moving the router's power supply switch to the Off position. However, any changes to the running config since the last WRITE of the config to the NVRAM is lost.

Ensure that any configuration needed after startup is saved before powering off the router. The copy running-config startup-config command saves the configuration in NVRAM and after the router is powered up, the router initializes with the saved configuration.

Finding Support Information for Platforms and Cisco Software Images

The Cisco IOS XE software is packaged in feature sets consisting of software images that support specific platforms. The group of feature sets that are available for a specific platform depends on which Cisco software images are included in a release. To identify the set of software images available in a specific release or to find out if a feature is available in a given Cisco IOS XE software image, you can use Cisco Feature Navigator or see the Release Notes for Cisco IOS XE.

Using Cisco Feature Navigator

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator is a tool that enables you to determine which Cisco IOS XE software images support a specific software release, feature set, or platform. To use the navigator tool, an account on Cisco.com is not required.

Using Software Advisor

Cisco maintains the Software Advisor tool. See Tools and Resources. Use the Software Advisor tool to see if a feature is supported in a Cisco IOS XE release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS XE software with the hardware installed on your router. You must be a registered user on Cisco.com to access this tool.

Using Software Release Notes

See the Release Notes document for the Cisco 4000 Series ISRs for information about the following:

- Memory recommendations
- Open and resolved severity 1 and 2 caveats

Release notes are intended to be release-specific for the most current release, and the information provided in these documents may not be cumulative in providing information about features that first appeared in previous releases. For cumulative feature information, refer to the Cisco Feature Navigator at: http://www.cisco.com/go/cfn/.

CLI Session Management

An inactivity timeout is configurable and can be enforced. Session locking provides protection from two users overwriting changes that the other has made. To prevent an internal process from using all the available capacity, some spare capacity is reserved for CLI session access. For example, this allows a user to remotely access a router.

- Changing the CLI Session Timeout, on page 59
- Locking a CLI Session, on page 59

Information About CLI Session Management

An inactivity timeout is configurable and can be enforced. Session locking provides protection from two users overwriting changes that each other has made. To prevent an internal process from using all the available capacity, some spare capacity is reserved for CLI session access. For example, this allows a user to remotely access the router.

Changing the CLI Session Timeout

Step 1 `configure terminal`

Enters global configuration mode

Step 2 `line console 0`

Step 3 **`session-timeout`** *`minutes`*

The value of *`minutes`* sets the amount of time that the CLI waits before timing out. Setting the CLI session timeout increases the security of a CLI session. Specify a value of 0 for *`minutes`* to disable session timeout.

Step 4 `show line console 0`

Verifies the value to which the session timeout has been set, which is shown as the value for " `Idle Session` ".

Locking a CLI Session

Before you begin

To configure a temporary password on a CLI session, use the **lock** command in EXEC mode. Before you can use the **lock** command, you need to configure the line using the **lockable** command. In this example the line is configured as **lockable**, and then the **lock** command is used and a temporary password is assigned.

Step 1 `Router# configure terminal`

Enters global configuration mode.

Step 2 Enter the line upon which you want to be able to use the **lock** command.

`Router(config)# line console 0`

Step 3 `Router(config)# lockable`

Enables the line to be locked.

Step 4 `Router(config)# exit`

Step 5 `Router# lock`

The system prompts you for a password, which you must enter twice.

```
Password: <password>
Again: <password>
Locked
```

CHAPTER 5

Smart Licensing

This chapter provides an overview of the Cisco Smart Licensing Client feature and describes the several tools and processes required to complete the products registration and authorization.

This chapter includes this section:

Introduction to Smart Licensing

Cisco Smart Licensing is a flexible licensing model that provides you with an easier, faster, and more consistent way to purchase and manage software across the Cisco portfolio and across your organization. And it's secure – you control what users can access. With Smart Licensing, you get:

- Easy Activation: Smart Licensing establishes a pool of software licenses that can be used across the entire organization—no more PAKs (Product Activation Keys).
- Unified Management: My Cisco Entitlements (MCE) provides a complete view into all of your Cisco products and services in an easy-to-use portal, so you always know what you have and what you are using.
- License Flexibility: Your software is not node-locked to your hardware, so you can easily use and transfer licenses as needed.

To use Smart Licensing, you must first set up a Smart Account on Cisco Software Central (http://software.cisco.com/).

For a more detailed overview on Cisco Licensing, go to https://cisco.com/go/licensingguide.

For Smart Licensing configuration information for access and edge routers, see the https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/smart-licensing/qsg/b_Smart_Licensing_QuickStart/b_Smart_Licensing_QuickStart_chapter_01.html.

Prerequisites for Cisco Smart Licensing Client

- Ensure that Call Home is enabled before using the Smart Licensing Client feature.
- Ensure that the device is running the Cisco IOS XE Everest 16.6.1 version that supports the Smart Licensing mode.

Restrictions for Cisco Smart Licensing Client

- Cisco 4000 Series ISR platforms support Cisco One Suites License, Technology Package License, Throughput License, and HSECK9 license in Cisco Smart Licensing from Cisco IOS XE Release 16.6.1.

Information About Cisco Smart Licensing Client

Cisco Smart Licensing - An Overview

Smart licensing has the capability to capture a customer's order and to communicate with Cisco Cloud License Service through the Smart Call Home Transport Gateway. Additionally, the Smart Call Home Transport Gateway helps to complete product registration and authorization based on the desired performance and technology levels of Cisco products. To know more about Call Home, refer to *Call Home* .

Benefits of Smart Licensing are the following:

- Support for CiscoONE suites in the Cisco IOS Software License (CISL) and Smart Licensing mode, including the Foundation Suite and Active Directory Users and Computers (ADUC) Suite.
- The ability to switch between traditional licensing (CSL) and Smart Licensing mode
- Support for four software universal images NPE, NO-LI, NPE-NO-LI, and Non-NPE images.

Transitioning from CSL to Smart Licensing

In the Smart Licensing Model, customers can activate licensed objects without the use of a special software key or upgrade license file. Customers simply activate the new functionality using the appropriate product commands and configurations and the functionality is activated. A software reboot may or may not be required depending on the product capabilities and requirements.

Similarly, downgrading or removing an advanced feature, performance, or functionality would require removal of the configuration or command.

After either of the above actions has been taken, the change in license state is noted by the Smart Software Manager upon next synchronization and an appropriate action is taken.

Cisco ONE Suites

Cisco ONE Suites is a new way for customers to purchase infrastructure software. Cisco ONE offers a simplified purchasing model, centered on common customer scenarios in the data center, wide area network, and local access networks.

Smart Licensing supports Smart License Cisco ONE suite level licenses and image licenses, such as ipbase, Advanced IP Services (AIS), Advanced Enterprise Services (AES) and feature license and throughput performance, crypto throughput and port licensing.

To know more about Cisco One Suites, please refer to Cisco ONE Suites.

How to Activate Cisco Smart Licensing Client

Enable Smart Licensing

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **license smart enable**
4. **exit**
5. **write memory**
6. **show license all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**license smart enable** **Example:** `Device# license smart enable`	Activates Smart Licensing on the device. **Note** When you enable Smart Licensing, the Cisco Software License (CSL) and all licensing calls pass through the Smart Agent. For the 'no' case, if Smart Licensing is already registered, the Smart Agent performs the "license smart deregister" operation that deactivates Smart Licensing. Reload the device to activate the CSL on the device.
Step 4	**exit** **Example:** `Device# exit`	Exits the global configuration mode.
Step 5	**write memory** **Example:** `Device# write memory`	Saves the running configuration to NVRAM.

	Command or Action	Purpose
Step 6	**show license all** **Example:** `Device# show license all`	(Optional) Displays summary information about all licenses.

Smart License Disable

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no license smart enable**
4. **exit**
5. **write memory**
6. **reload**
7. **show license all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**no license smart enable** **Example:** `Device(config)# no license smart enable`	Deactivates Smart Licensing on the device. **Note** When you enable Smart Licensing, the Cisco Software License (CSL) and all licensing calls pass through the Smart Agent. For the 'no' case, if Smart Licensing is already registered, the Smart Agent performs the "license smart deregister" operation that deactivates Smart Licensing. Reload the device to activate the CSL on the device.
Step 4	**exit** **Example:** `Device(config)# exit`	Exits the global configuration mode.

	Command or Action	Purpose
Step 5	**write memory** **Example:** `Device# write memory`	Saves the running configuration to NVRAM.
Step 6	**reload** **Example:** `Device# reload`	(Optional) Restarts the device to enable the new feature set. **Note** Reload the device if you have not reloaded the device after configuring the Cisco One Suites.
Step 7	**show license all** **Example:** `Device# show license all`	(Optional) Displays summary information about all licenses.

Device Registration

SUMMARY STEPS

1. **enable**
2. **license smart register idtoken** *idtoken* [**force**]
3. **license smart deregister**
4. **license smart renew** [**ID** | **auth**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**license smart register idtoken** *idtoken* [**force**] **Example:** `Device# license smart register idtoken 123`	Registers the device with the back-end server. Token id can be obtained from your virtual a/c in the Smart Licensing server. • **force**: To forcefully register your device irrespective of either the device is registered or not. **Note** The device supplies the token ID to the Cisco server, which sends back a "Device Certificate" that is valid for 365 days.
Step 3	**license smart deregister** **Example:** `Device# license smart deregister`	Deregisters the device from the backend server.

	Command or Action	Purpose
Step 4	**license smart renew** [**ID** | **auth**] **Example:** `Device# license smart renew ID`	(Optional) Manually renews the ID certification or authorization. For more information on license boot level, license feature hseck9, and platform hardware throughput level, see the Smart Licensing Guide for Access and Edge Routers.

Troubleshooting for Cisco Smart Licensing Client

You can troubleshoot Smart Licensing enabling issues using the following commands on the device:

- **show version**
- **show running-config**
- **show license summary**
- **show license all**
- **show license tech support**
- **debug smart_lic error**
- **debug smart_lic trace**

Configuration Examples for Cisco Smart Licensing Client

Example: Displays summary information about all licenses

The following example shows how to use the **show license all** command to display summary information about all licenses.

```
Device#show license all
Smart Licensing Status
======================

Smart Licensing is ENABLED

Registration:
Status: REGISTERED
Smart Account: BU Production Test
Virtual Account: ISR4K
Export-Controlled Functionality: Allowed
Initial Registration: SUCCEEDED on Sep 04 15:40:03 2015 PDT
Last Renewal Attempt: None
Next Renewal Attempt: Mar 02 15:40:02 2016 PDT
Registration Expires: Sep 03 15:34:53 2016 PDT

License Authorization:
Status: AUTHORIZED on Sep 04 15:40:09 2015 PDT
Last Communication Attempt: SUCCEEDED on Sep 04 15:40:09 2015 PDT
Next Communication Attempt: Oct 04 15:40:08 2015 PDT
Communication Deadline: Dec 03 15:35:01 2015 PDT

License Usage
```

```
==============

ISR_4400_FoundationSuite (ISR_4400_FoundationSuite):
Description: Cisco ONE Foundation Perpetual License ISR 4400
Count: 1
Version: 1.0
Status: AUTHORIZED

ISR_4400_AdvancedUCSuite (ISR_4400_AdvancedUCSuite):
Description: Cisco ONE Advanced UC Perpetual License ISR 4400
Count: 1
Version: 1.0
Status: AUTHORIZED

ISR_4451_2G_Performance (ISR_4451_2G_Performance):
Description: Performance on Demand License for 4450 Series
Count: 1
Version: 1.0
Status: AUTHORIZED

Product Information
===================
UDI: PID:ISR4451-X/K9,SN:FOC17042FJ9

Agent Version
=============
Smart Agent for Licensing: 1.4.0_rel/16
Component Versions: SA:(1_4_rel)1.0.15, SI:(dev22)1.2.6, CH:(dev5)1.0.32, PK:(dev18)1.0.17


Device#
```

Example: Enabling Smart Licensing

The following example shows how to use the **license smart enable** command to confirm if the Cisco ONE Suite is enabled.

Note The warning message that is displayed in the following example applies only for Cisco ISR G2 platform. For Cisco 4000 Series ISR platform, it does not display warning message when you enable the smart license.

```
Device# license smart enable
Currently only Cisco ONE license suites are supported by Smart Licensing.
Please make sure your Cisco ONE suites are enabled before turning on Smart Licensing.
Any other licenses outside of Cisco ONE suites would be disabled and made unusable in Smart
 Licensing.
If you have any questions, please get in touch with your Cisco representative before using
 this mmode.
Please confirm Cisco ONE suites are enabled? [yes/no]: yes
```

CHAPTER 6

Managing the Device Using Web User Interface

The Web User Interface (Web UI) is an embedded GUI-based device-management tool that provides the ability to provision the device, to simplify device deployment and manageability, and to enhance the user experience. It comes with the default image, so there is no need to enable anything or install any license on the device. You can use Web UI to build configurations, and to monitor and troubleshoot the device without having CLI expertise. This chapter includes the following sections:

Setting Up Factory Default Device Using WebUI

Quick Setup Wizard allows you to perform the basic router configuration. To configure the router:

Before you begin

- Before you access the WebUI, you need to have the basic configuration on the device.

Step 1 Connect the RJ-45 end of a serial cable to the RJ-45 console port on the router.

Step 2 After the device initial configuration wizard appears, enter **No** to get into the device prompt when the following system message appears on the router.

Would you like to enter the initial configuration dialog? [yes/no]: no

Step 3 From the configuration mode, enter the following configuration parameters.

```
!
ip dhcp pool WEBUIPool
network 192.168.1.0 255.255.255.0
default-router 192.168.1.1
username webui privilege 15 password cisco
!
interface gig 0/0/1
ip address 192.168.1.1 255.255.255.0
!
```

Step 4 Connect your device to the router using an Ethernet cable to the gig 0/0/1 interface.

Step 5 Set up your system as a DHCP client to obtain the IP address of the router automatically.

Step 6 Launch the browser and enter the device IP address in your browser's address line. For a secure connection, type https://192.168.1.1/#/dayZeroRouting. For a less secure connection, enter http://192.168.1.1/#/dayZeroRouting.

Step 7 Enter the default username (webui) and default password (cisco).

Using Basic or Advanced Mode Setup Wizard

To configure the router using the basic or advanced mode setup:

Step 1 Choose the **Basic Mode** or **Advanced Mode** and click **Go To Account Creation Page**.

Step 2 Enter the username and password. Reenter the password to confirm.

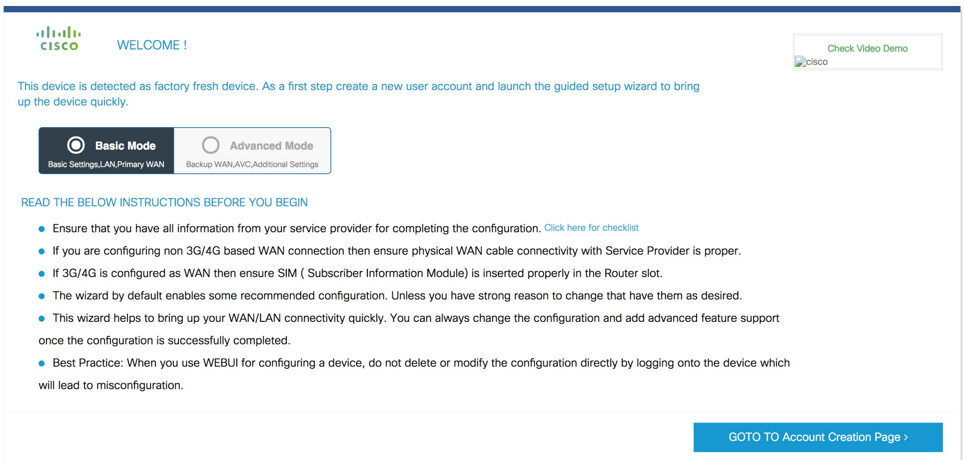
Step 3 Click **Create and Launch Wizard**.

Step 4 Enter the device name and domain name.

Step 5 Select the appropriate time zone from the **Time Zone** drop-down list.

Step 6 Select the appropriate date and time mode from the **Date and Time** drop-down list.

Step 7 Click **LAN Settings**.



Configure LAN Settings

Step 1 Choose the **Web DHCP Pool/DHCP Pool** name or the **Create and Associate Access VLAN** option.

a) If you choose the Web DHCP Pool, specify the following:

Pool Name—Enter the DHCP Pool Name.

Network—Enter network address and the subnet mask.

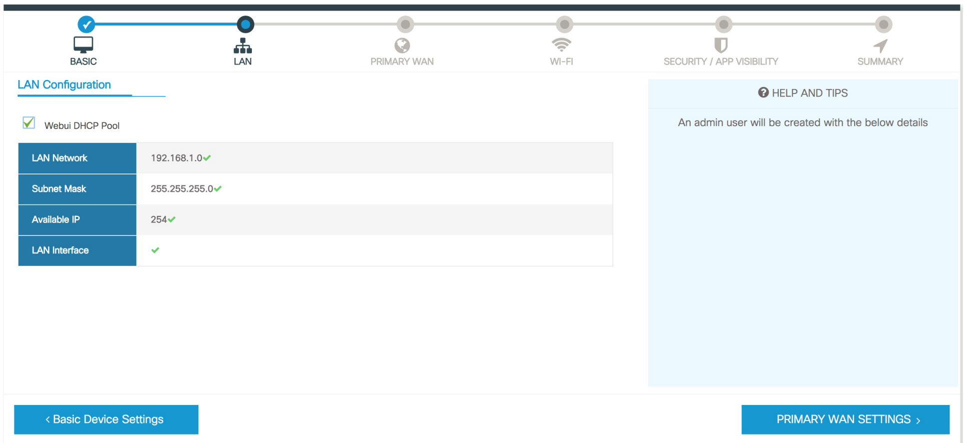
b) If you choose the Create and Associate Access VLAN option, specify the following:

Access VLAN—Enter the Access VLAN identification number. The range is from 1 to 4094.

Network—Enter the IP address of the VLAN.

Management Interfaces—Select the interface and move to the selected list box using the right and left arrows. You can also double click or drag and drop to move the interface to the selected list box.

Step 2 Click **Primary WAN Settings**.



Configure Primary WAN Settings

Step 1 Select the primary WAN type. You can configure Serial, 3G/4G, Ethernet, or Broadband (xDSL) as primary WAN depending on the WAN types supported by the router.

Step 2 Select the interface from the drop-down list.

Step 3 Check the **Get DNS Server info directly from ISP** check box to get the DNS server information directly from the service provider. You can also manually enter the Primary DNS and Secondary DNS.

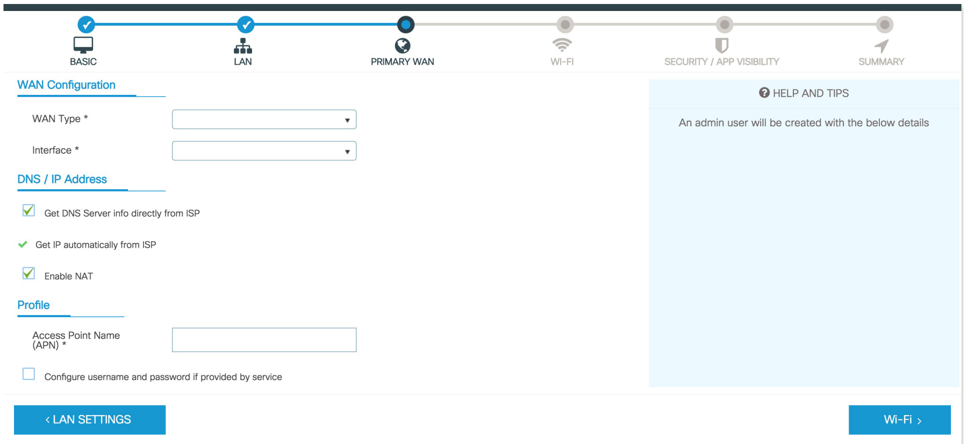
Step 4 Check the **Get IP automatically from ISP** check box to get the IP address information directly from the service provider. You can also manually enter the IP address and subnet mask.

Step 5 Check the **Enable NAT** check box to enable NAT. It is recommended to enable NAT.

Step 6 Check the **Enable PPPOE** check box to enable PPPoE. If you have enabled PPPoE, select the required authentication mode. The options are: **PAP** and **CHAP**.

Step 7 Enter the username and password provided by the service provider.

Step 8 Click **Security / APP Visibility WAN Settings**.



Configure Secondary WAN Settings

For advanced configuration, you should configure the secondary WAN connection.

Step 1 Select the secondary WAN type. You can configure Serial, 3G/4G, Ethernet, or Broadband (xDSL) as a secondary WAN depending on the WAN types supported by the router.

Step 2 Select the interface from the drop-down list.

Step 3 Check the **Get DNS Server info directly from ISP** check box to get the DNS server information directly from the service provider. You can also manually enter the Primary DNS and Secondary DNS.

Step 4 Check the **Get IP automatically from ISP** check box to get the IP address information directly from the service provider. You can also manually enter the IP address and subnet mask.

Step 5 Check the **Enable NAT** check box to enable NAT. It is recommended to enable NAT.

Step 6 Check the **Enable PPPOE** check box to enable PPPoE. If you have enabled PPPoE, select the required authentication mode. The options are **PAP** and **CHAP**.

Step 7 Enter the username and password provided by the service provider.

Step 8 Click **Security / APP Visibility WAN Settings**.

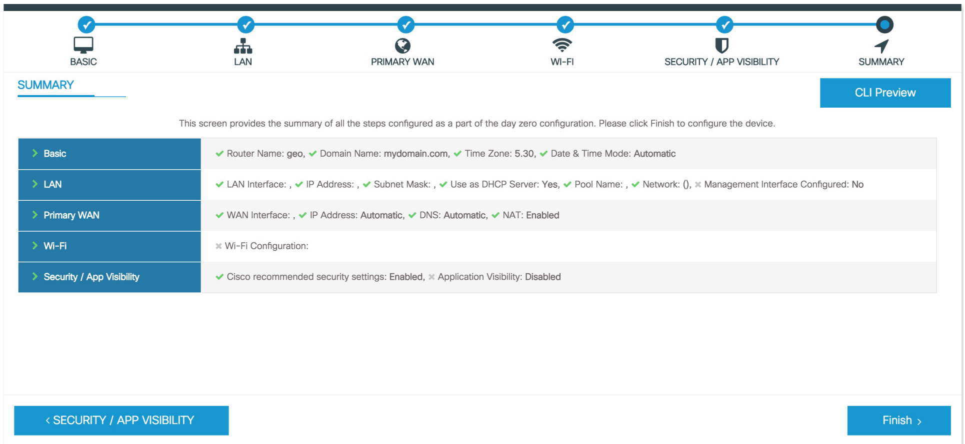
Configure Security Settings

Step 1 Check the **Enable Cisco Recommended Security Settings** check box to ensure that all passwords are not shown in plain text. The passwords are encrypted.

Step 2 Click **Day 0 Config Summary**.

Step 3 To preview the configuration, click **CLI Preview** to preview the configuration.

Step 4 Click **Finish** to complete the Day Zero setup.



Using Web User Interface for Day One Setup

To configure the Web user interface:

Before you begin

- You need to configure at least 30 VTY lines on the device for the Web UI information to be displayed without errors.
- You need a user with privilege 15 to access the configuration screens on Web UI. If the privilege is less than 15, you can access only the Dashboard and Monitoring screens on Web UI.

  To create a user account, use the **username** <username> **privilege** <privilege> **password 0** <passwordtext>

  ```
  Device #configure terminal
  Device (config)# username <username> privilege <privilege> password 0
  <passwordtext>
  ```

Step 1 Configure the HTTP server. By default, the HTTP server configuration should be present on the device. Ensure the configuration by checking if the **ip http server** and **ip http secure-server** commands are present in the running configuration.

```
Device #configure terminal
Device (config)#ip http server
Device (config)#ip http secure-server
```

Step 2 Set up the authentication options to log into Web UI. You can use one of these methods to authenticate:

a) You can authenticate using local database. To use a local database for Web UI authentication, ensure to have the **ip http authentication local** command in the running configuration. This command is preconfigured on the device. If the command is not present, configure the device as shown in this example:

```
Device #configure terminal
 Device (config)#ip http authentication local
```

b) Authenticate using AAA options. To use AAA authentication for Web UI, ensure to configure ‘ip http authentication aaa’ on the device. Also, ensure that the required AAA server configuration is present on the device.

```
Device #configure terminal
Device (config)#ip http authentication local
```

Step 3 Launch the browser. In the address bar, type the IP address of the device. For a secure connection, type https://ip-address.

Step 4 Enter the default username (webui) and default password (cisco).

Step 5 Click **Log In**.

Monitor and Troubleshoot Device Plug and Play (PnP) Onboarding using WebUI

Table 11: Feature History

Feature Name	Release Information	Description
Monitor and Troubleshoot Device PnP Onboarding using WebUI	Cisco IOS XE Catalyst SD-WAN Release 17.5.1a	You can now monitor and troubleshoot your Day-0 device onboarding using WebUI through PnP onboarding. If the automated PnP onboarding fails, you can manually onboard your device.

A device can be automatically onboarded to Cisco vManage through either Zero Touch Provisioning (ZTP) or the Plug and Play (PnP) process. This section describes the procedure to monitor and troubleshoot device onboarding through the PnP method. This feature on WebUI enables you to monitor and troubleshoot the PnP onboarding process, and also see its real-time status. If this onboarding is stuck or fails, you can terminate the process and onboard your device manually.

Prerequisites

- Your device (a computer that can run a web browser) running the WebUI and the device you are onboarding must be connected through an L2 switch port (NIM) on the device.
- The DHCP client-identifier on your device must be set to string “webui”.
- Your device must support Cisco SD-WAN Day-0 device onboarding on WebUI.

Troubleshoot Device PnP Onboarding

To troubleshoot device onboarding through PnP in controller mode:

1. Enter the controller mode in WebUI:

    - Switching from autonomous mode to controller mode:

      Usually, when you boot your device for the first time it is in autonomous mode. Go to the URL https://192.168.1.1/webui/ and log in using the default credentials— webui/cisco. If your device supports Cisco SD-WAN Day-0 device onboarding on WebUI, you can switch to the controller mode by selecting **Controller Mode.** A dialogue box appears, asking if you want to continue. Click **Yes.** Your device reloads to switch to controller mode.

- Booting your device in controller mode:

  If your device is already in the controller mode, you do not have to make any changes to the mode. Go to the URL https://192.168.1.1 or https://192.168.1.1/webui. If your device supports Cisco SD-WAN Day-0 device onboarding on WebUI, the URL is redirected to https://192.168.1.1/ciscosdwan/ and you can log in using the default credentials for Cisco IOS XE SD-WAN devices - admin/admin.

Note If the device does not have start-up configuration at the time of PnP onboarding, the WebUI is enabled by default on supported devices.

2. On the **Welcome to Cisco SDWAN Onboarding Wizard** page, click **Reset Default Password.**

Note The default password of your Day-0 device is weak. Therefore, for a secure log in, you must reset the password when you first log in to the device on WebUI. The WebUI configuration is automatically deleted after the device is onboarded successfully. In rare cases where the template configuration for your device on Cisco vManage has the WebUI configuration, it is not deleted even after a successful device onboarding.

3. You are redirected to the Device hardware and software details page. Enter your password and click **Submit.**

4. The next page displays the onboarding progress and lists statuses of different components of the PnP Connect Portal and Cisco SD-WAN controllers. If the PnP IPv4 component fails, it indicates that the device PnP onboarding has failed.

   To view and download logs for the onboarding process, click the information icon on the right hand side of the SDWAN Onboarding Progress bar.

5. If the automated PnP onboarding fails, click **Terminate Automated Onboarding.** This allows you to onboard your device manually.

6. A dialogue box appears. To continue with the termination, click **Yes**. It might take a few minutes for the termination to complete.

7. On the Bootstrap Configuration page click **Select File** and choose the bootstrap file for your device. This file can be either a generic bootstrap file (common platform-specific file) or a full configuration bootstrap file that you can download from Cisco SD-WAN Manager. This file must contain details such as the vBond number, UUID, WAN interface, root CA and configuration.

8. Click **Upload**.

9. After your file is successfully uploaded, click **Submit.**

10. You can see the SDWAN Onboarding Progress page again with statuses of the Cisco SD-WAN controllers. To open the Controller Connection History table click the information icon on the right hand side of the SDWAN Control Connections bar. In this table you can see the state of your onboarded device. After the onboarding is complete, the state of your device changes to **connect**.

CHAPTER 7

Console Port, Telnet, and SSH Handling

This chapter includes the following sections:

Notes and Restrictions for Console Port, Telnet, and SSH

- Telnet and Secure Shell (SSH) settings configured in the transport map override any other Telnet or SSH settings when the transport map is applied to the Ethernet management interface.
- Only local usernames and passwords can be used to authenticate users entering a Ethernet management interface. AAA authentication is not available for users accessing the router through a Ethernet management interface using persistent Telnet or persistent SSH.
- Applying a transport map to a Ethernet management interface with active Telnet or SSH sessions can disconnect the active sessions. Removing a transport map from an interface, however, does not disconnect any active Telnet or SSH session.
- Configuring the diagnostic and wait banners is optional, but recommended. The banners are especially useful as indicators to users about the status of their Telnet or SSH attempts.

Console Port Overview

The console port on the router is an EIA/TIA-232 asynchronous, serial connection with no flow control and an RJ-45 connector. The console port is used to access the router and is located on the front panel of the Route Processor.

For information on accessing the router using the console port, see Using Cisco IOS XE Software, on page 45.

Console Port Handling Overview

If you are using the console port to access the router, you are automatically directed to the Cisco IOS command-line interface (CLI).

If you are trying to access the router through the console port and send a break signal (by entering **Ctrl-C** or **Ctrl-Shift-6**, or by entering the **send break** command at the Telnet prompt) before connecting to the CLI, you are directed to a diagnostic mode if the non-RPIOS subpackages are accessible. These settings can be changed by configuring a transport map for the console port and applying that transport map to the console interface.

Telnet and SSH Overview

Telnet and SSH on the router can be configured and handled like Telnet and SSH on other Cisco platforms. For information on traditional Telnet, see the line command in the Cisco IOS Terminal Services Command Reference, Release 12.2 document. For more information on AAA authentiction methods, see the line command in the Authentication Commands chapter.

For information on configuring traditional SSH, see the "Configuring Secure Shell" chapter in the Cisco IOS Terminal Services Command Reference, Release 12.2 document.

On the router, persistent Telnet and persistent SSH allow network administrators to more clearly define the treatment of incoming traffic when users access the router through the management ethernet port using Telnet or SSH. Notably, persistent Telnet and persistent SSH provide more robust network access by allowing the router to be configured to be accessible through the Ethernet management port using Telnet or SSH even when the Cisco IOS process has failed.

Persistent Telnet and Persistent SSH Overview

In traditional Cisco routers, accessing the router using Telnet or SSH is not possible if the Cisco IOS software fails. When Cisco IOS fails on a traditional Cisco router, the only method of accessing the router is through the console port. Similarly, if all the active Cisco IOS processes have failed on a router that is not using persistent Telnet or persistent SSH, the only method of accessing the router is through the console port.

However, with persistent Telnet and persistent SSH, you can configure a transport map that defines the treatment of incoming Telnet or SSH traffic on the Ethernet management interface. Among the many configuration options, a transport map can be configured to direct all traffic to the Cisco IOS CLI, diagnostic mode, or to wait for a Cisco IOS VTY line to become available and then direct users to diagnostic mode when a user sends a break signal while waiting for the IOS VTY line to become available. If a user uses Telnet or SSH to access diagnostic mode, that Telnet or SSH connection will be usable even in scenarios when no Cisco IOS process is active. Therefore, persistent Telnet and persistent SSH introduce the ability to access the router via diagnostic mode when the Cisco IOS process is not active. For information on diagnostic mode, see Using Cisco IOS XE Software. For information on the options that are can be configured using persistent Telnet or persistent SSH transport maps, see Configuring Persistent Telnet, on page 81 and Configuring Persistent SSH, on page 83.

Configuring a Console Port Transport Map

This task describes how to configure a transport map for a console port interface on the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **transport-map type console** *transport-map-name*
4. **connection wait** [**allow** [**interruptible**] | **none** [**disconnect**]]
5. (Optional) **banner** [**diagnostic** | **wait**] *banner-message*
6. **exit**
7. **transport type console** *console-line-number* **input** *transport-map-name*

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	**transport-map type console** *transport-map-name* **Example:** `Router(config)# transport-map type console consolehandler`	Creates and names a transport map for handling console connections, and enters transport map configuration mode.
Step 4	**connection wait** [**allow** [**interruptible**] | **none** [**disconnect**]] **Example:** `Router(config-tmap)# connection wait none`	Specifies how a console connection will be handled using this transport map. • **allow interruptible**—The console connection waits for a Cisco IOS VTY line to become available, and also allows users to enter diagnostic mode by interrupting a console connection that is waiting for a Cisco IOS VTY line to become available. This is the default setting. **Note** Users can interrupt a waiting connection by entering **Ctrl-C** or **Ctrl-Shift-6**. • **none**—The console connection immediately enters diagnostic mode.

	Command or Action	Purpose

Step 5	(Optional) **banner** [**diagnostic** | **wait**] *banner-message*
	Example:
	`Router(config-tmap)# banner diagnostic X`
	`Enter TEXT message. End with the character 'X'.`
	`--Welcome to Diagnostic Mode--`
	`X`
	`Router(config-tmap)#`
		(Optional) Creates a banner message that will be seen by users entering diagnostic mode or waiting for the Cisco IOS VTY line because of the console transport map configuration.
		• **diagnostic**—Creates a banner message seen by users directed to diagnostic mode because of the console transport map configuration.
		Note Users can interrupt a waiting connection by entering **Ctrl-C** or **Ctrl-Shift-6**.
		• **wait**—Creates a banner message seen by users waiting for Cisco IOS VTY to become available.
		• *banner-message*—Banner message, which begins and ends with the same delimiting character.

Step 6	**exit** **Example:** `Router(config-tmap)# exit`	Exits transport map configuration mode to re-enter global configuration mode.
Step 7	**transport type console** *console-line-number* **input** *transport-map-name* **Example:** `Router(config)# transport type console 0 input consolehandler`	Applies the settings defined in the transport map to the console interface. The *transport-map-name* for this command must match the *transport-map-name* defined in the **transport-map type console** command.

Examples

The following example shows how to create a transport map to set console port access policies and attach to console port 0:

```
Router(config)# transport-map type console consolehandler
Router(config-tmap)# connection wait allow interruptible
Router(config-tmap)# banner diagnostic X
Enter TEXT message. End with the character 'X'.
--Welcome to diagnostic mode--
X
Router(config-tmap)# banner wait X
Enter TEXT message. End with the character 'X'.
Waiting for IOS vty line
X
Router(config-tmap)# exit
Router(config)# transport type console 0 input consolehandler
```

Configuring Persistent Telnet

For a persistent Telnet connection to access an Cisco IOS vty line on the router, local login authentication must be configured for the vty line (the **login** command in line configuration mode). If local login authentication is not configured, users will not be able to access Cisco IOS using a Telnet connection into the management Ethernet interface with an applied transport map. Diagnostic mode will still be accessible in this scenario.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **transport-map type persistent telnet** *transport-map-name*
4. **connection wait** [**allow** [**interruptible**] | **none** [**disconnect**]]
5. (Optional) **banner** [**diagnostic** | **wait**] *banner-message*
6. **transport interface gigabitethernet 0**
7. **exit**
8. **transport type persistent telnetinput** *transport-map-name*

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	**transport-map type persistent telnet** *transport-map-name* **Example:** `Router(config)# transport-map type persistent telnet telnethandler`	Creates and names a transport map for handling persistent Telnet connections, and enters transport map configuration mode.
Step 4	**connection wait** [**allow** [**interruptible**] | **none** [**disconnect**]] **Example:** `Router(config-tmap)# connection wait none`	Specifies how a persistent Telnet connection will be handled using this transport map: • **allow**—The Telnet connection waits for a Cisco IOS vty line to become available, and exits the router if interrupted. • **allow interruptible**—The Telnet connection waits for the Cisco IOS vty line to become available, and also allows user to enter diagnostic mode by interrupting

	Command or Action	Purpose
		a Telnet connection waiting for the Cisco IOS vty line to become available. This is the default setting. **Note** Users can interrupt a waiting connection by entering **Ctrl-C** or **Ctrl-Shift-6**. • **none**—The Telnet connection immediately enters diagnostic mode. • **none disconnect**—The Telnet connection does not wait for the Cisco IOS vty line and does not enter diagnostic mode, so all Telnet connections are rejected if no vty line is immediately available in the Cisco IOS software.

Step 5	(Optional) **banner** [**diagnostic** | **wait**] *banner-message*
	Example:
	`Router(config-tmap)# banner diagnostic X`
	`Enter TEXT message. End with the character 'X'.`
	`--Welcome to Diagnostic Mode--`
	`X`
	`Router(config-tmap)#`
		(Optional) Creates a banner message that will be seen by users entering diagnostic mode or waiting for the Cisco IOS vty line because of the persistent Telnet configuration.
		• **diagnostic**—Creates a banner message seen by users directed into diagnostic mode because of the persistent Telnet configuration.
		Note Users can interrupt a waiting connection by entering **Ctrl-C** or **Ctrl-Shift-6**.
		• **wait**—Creates a banner message seen by users waiting for the vty line to become available.
		• *banner-message*—The banner message, which begins and ends with the same delimiting character.

Step 6	**transport interface gigabitethernet 0** **Example:** `Router(config-tmap)# transport interface gigabitethernet 0`	Applies the transport map settings to the management Ethernet interface (interface gigabitethernet 0). Persistent Telnet can be applied only to the management Ethernet interface on the router. This step must be taken before applying the transport map to the management Ethernet interface.
Step 7	**exit** **Example:** `Router(config-tmap)# exit`	Exits transport map configuration mode to re-enter global configuration mode.
Step 8	**transport type persistent telnetinput** *transport-map-name* **Example:** `Router(config)# transport type persistent telnet input telnethandler`	Applies the settings defined in the transport map to the management Ethernet interface. The *transport-map-name* for this command must match the *transport-map-name* defined in the **transport-map type persistent telnet** command.

Examples

In the following example, a transport map that will make all Telnet connections wait for a Cisco IOS XE vty line to become available before connecting to the router, while also allowing the user to interrupt the process and enter diagnostic mode, is configured and applied to the management Ethernet interface (**interface gigabitethernet 0**).

A diagnostic and a wait banner are also configured.

The transport map is then applied to the interface when the **transport type persistent telnet input** command is entered to enable persistent Telnet.

```
Router(config)# transport-map type persistent telnet telnethandler
Router(config-tmap)# connection wait allow interruptible
Router(config-tmap)# banner diagnostic X
Enter TEXT message. End with the character 'X'.
--Welcome to diagnostic mode--
X
Router(config-tmap)# banner wait X
Enter TEXT message. End with the character 'X'.
Waiting for IOS IOS Process--
X
Router(config-tmap)# transport interface gigabitethernet 0
Router(config-tmap)# exit
Router(config)# transport type persistent telnet input telnethandler
```

Configuring Persistent SSH

This task describes how to configure persistent SSH on the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **transport-map type persistent ssh** *transport-map-name*
4. **connection wait** [**allow** [**interruptible**] | **none** [**disconnect**]]
5. **rsa keypair-name** *rsa-keypair-name*
6. (Optional) **authentication-retries** *number-of-retries*
7. (Optional) **banner** [**diagnostic** | **wait**] *banner-message*
8. (Optional) **time-out** *timeout-interval*
9. **transport interface gigabitethernet 0**
10. **exit**
11. **transport type persistent ssh input** *transport-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	`Router> enable`	
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**transport-map type persistent ssh** *transport-map-name* **Example:** `Router(config)# transport-map type persistent telnet telnethandler`	Creates and names a transport map for handling persistent SSH connections, and enters transport map configuration mode.
Step 4	**connection wait** [**allow** [**interruptible**] | **none** [**disconnect**]] **Example:** `Router(config-tmap)# connection wait interruptible`	Specifies how a persistent SSH connection will be handled using this transport map: • **allow**—The SSH connection waits for a Cisco IOS VTY line to become available, and exits the router if interrupted. • **allow interruptible**—The SSH connection waits for the VTY line to become available, and also allows a user to enter diagnostic mode by interrupting an SSH connection waiting for the VTY line to become available. This is the default setting. **Note** Users can interrupt a waiting connection by entering **Ctrl-C** or **Ctrl-Shift-6**. • **none**—The SSH connection immediately enters diagnostic mode. • **none disconnect**—The SSH connection does not wait for the VTY line and does not enter diagnostic mode. Therefore, all SSH connections are rejected if no VTY line is immediately available.
Step 5	**rsa keypair-name** *rsa-keypair-name* **Example:** `Router(config)# rsa keypair-name sshkeys`	Names the RSA keypair to be used for persistent SSH connections. For persistent SSH connections, the RSA keypair name must be defined using this command in transport map configuration mode. The RSA keypair definitions defined elsewhere on the router, such as through the use of the **ip ssh rsa keypair-name** command, do not apply to persistent SSH connections. No *rsa-keypair-name* is defined by default.

	Command or Action	Purpose
Step 6	(Optional) **authentication-retries** *number-of-retries* **Example:** `Router(config-tmap)# authentication-retries 4`	(Optional) Specifies the number of authentication retries before dropping the connection. The default *number-of-retries* is 3.

Step 7	(Optional) **banner** [**diagnostic** | **wait**] *banner-message*
	Example:
	`Router(config-tmap)# banner diagnostic X`
	`Enter TEXT message. End with the character 'X'.`
	`--Welcome to Diagnostic Mode--`
	`X`
	`Router(config-tmap)#`
		(Optional) Creates a banner message that will be seen by users entering diagnostic mode or waiting for the VTY line because of the persistent SSH configuration.
		• **diagnostic**—Creates a banner message seen by users directed to diagnostic mode because of the persistent SSH configuration.
		• **wait**—Creates a banner message seen by users waiting for the VTY line to become available.
		• *banner-message*—The banner message, which begins and ends with the same delimiting character.

Step 8	(Optional) **time-out** *timeout-interval* **Example:** `Router(config-tmap)# time-out 30`	(Optional) Specifies the SSH time-out interval, in seconds. The default *timeout-interval* is 120 seconds.
Step 9	**transport interface gigabitethernet 0** **Example:** `Router(config-tmap)# transport interface gigabitethernet 0`	Applies the transport map settings to the Ethernet management interface (interface gigabitethernet 0). Persistent SSH can be applied only to the Ethernet management interface on the router.
Step 10	**exit** **Example:** `Router(config-tmap)# exit`	Exits transport map configuration mode to re-enter global configuration mode.
Step 11	**transport type persistent ssh input** *transport-map-name* **Example:** `Router(config)# transport type persistent ssh input sshhandler`	Applies the settings defined in the transport map to the Ethernet management interface. The *transport-map-name* for this command must match the *transport-map-name* defined in the **transport-map type persistent ssh** command.

Examples

The following example shows a transport map that will make all SSH connections wait for the VTY line to become active before connecting to the router being configured and applied to the Ethernet management interface (interface gigabitethernet 0). The RSA keypair is named `sshkeys`.

This example only uses the commands required to configure persistent SSH.

```
Router(config)# transport-map type persistent ssh sshhandler
Router(config-tmap)# connection wait allow
Router(config-tmap)# rsa keypair-name sshkeys
Router(config-tmap)# transport interface gigabitethernet 0
Enter TEXT message. End with the character 'X'.
--Welcome to diagnostic mode--
X
Router(config-tmap)# banner wait X
Enter TEXT message. End with the character 'X'.
--Waiting for IOS IOS Process--
X
Router(config-tmap)# transport interface gigabitethernet 0
Router(config-tmap)# exit
Router(config)# transport type persistent telnet input telnethandler
```

In the following example, a transport map is configured and will apply the following settings to users attempting to access the Ethernet management port via SSH:

- SSH users will wait for the VTY line to become active, but will enter diagnostic mode if the attempt to access the Cisco IOS software through the VTY line is interrupted.
- The RSA keypair name is sshkeys.
- The connection allows one authentication retry.
- The banner `--Welcome to Diagnostic Mode--` will appear if diagnostic mode is entered as a result of SSH handling through this transport map.
- The banner `--Waiting for vty line--` will appear if the connection is waiting for the VTY line to become active.
- The transport map is then applied to the interface when the **transport type persistent ssh input** command is entered to enable persistent SSH:

```
Router(config)# transport-map type persistent ssh sshhandler
Router(config-tmap)# connection wait allow interruptible
Router(config-tmap)# rsa keypair-name sshkeys
Router(config-tmap)# authentication-retries 1
Router(config-tmap)# banner diagnostic X
Enter TEXT message. End with the character 'X'.
--Welcome to diagnostic mode--
X
Router(config-tmap)# banner wait X
Enter TEXT message. End with the character 'X'.
--Waiting for vty line--
X
Router(config-tmap)# time-out 30
Router(config-tmap)# transport interface gigabitethernet 0
Router(config-tmap)# exit
Router(config)# transport type persistent ssh input sshhandler
```

Viewing Console Port, SSH, and Telnet Handling Configurations

Use the following commands to view console port, SSH, and Telnet handling configurations:

- **show transport-map**
- **show platform software configuration access policy**

Use the **show transport-map** command to view transport map configurations.

show transport-map [**all** | **name** *transport-map-name* | **type** [**console** | **persistent** [**ssh** | **telnet**]]]

This command can be used either in user EXEC mode or privileged EXEC mode.

Example

The following example shows transport maps that are configured on the router: a console port (`consolehandler`), persistent SSH (`sshhandler`), and persistent Telnet transport (`telnethandler`):

```
Router# show transport-map all
Transport Map:
Name: consolehandler
Type: Console Transport

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for the IOS CLI

bshell banner:

Welcome to Diagnostic Mode


Transport Map:
Name: sshhandler
Type: Persistent SSH Transport

Interface:
GigabitEthernet0

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for IOS prompt

Bshell banner:
Welcome to Diagnostic Mode


SSH:
Timeout: 120
Authentication retries: 5
RSA keypair: sshkeys

Transport Map:
Name: telnethandler
Type: Persistent Telnet Transport

Interface:
GigabitEthernet0

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for IOS process

Bshell banner:
```

```
Welcome to Diagnostic Mode


Transport Map:
Name: telnethandling1
Type: Persistent Telnet Transport

Connection:
Wait option: Wait Allow


Router# show transport-map type console
Transport Map:
Name: consolehandler
Type: Console Transport

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for the IOS CLI

Bshell banner:

Welcome to Diagnostic Mode


Router# show transport-map type persistent ssh
Transport Map:
Name: sshhandler
Type: Persistent SSH Transport

Interface:
GigabitEthernet0

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for IOS prompt

Bshell banner:

Welcome to Diagnostic Mode


SSH:
Timeout: 120
Authentication retries: 5
RSA keypair: sshkeys

Router# show transport-map type persistent telnet
Transport Map:
Name: telnethandler
Type: Persistent Telnet Transport

Interface:
GigabitEthernet0

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for IOS process
```

```
Bshell banner:

Welcome to Diagnostic Mode


Transport Map:
Name: telnethandling1
Type: Persistent Telnet Transport

Connection:
Wait option: Wait Allow


Router# show transport-map name telnethandler
Transport Map:
Name: telnethandler
Type: Persistent Telnet Transport

Interface:
GigabitEthernet0

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for IOS process

Bshell banner:

Welcome to Diagnostic Mode


Router# show transport-map name consolehandler
Transport Map:
Name: consolehandler
Type: Console Transport

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for the IOS CLI

Bshell banner:

Welcome to Diagnostic Mode


Router# show transport-map name sshhandler
Transport Map:
Name: sshhandler
Type: Persistent SSH Transport

Interface:
GigabitEthernet0

Connection:
Wait option: Wait Allow Interruptable
Wait banner:

Waiting for IOS prompt

Bshell banner:
```

```
Welcome to Diagnostic Mode


SSH:
Timeout: 120
Authentication retries: 5
RSA keypair: sshkeys

Router#
```

Use the **show platform software configuration access policy** command to view the current configurations for handling the incoming console port, SSH, and Telnet connections. The output of this command provides the current wait policy for each type of connection (Telnet, SSH, and console), as well as information on the currently configured banners.

Unlike the **show transport-map** command, the **show platform software configuration access policy** command is available in diagnostic mode so that it can be entered in scenarios where you need transport map configuration information, but cannot access the Cisco IOS CLI.

Example

```
Router# show platform software configuration access policy
The current access-policies

Method : telnet
Rule : wait
Shell banner:
Wait banner :

Method : ssh
Rule : wait
Shell banner:
Wait banner :

Method : console
Rule : wait with interrupt
Shell banner:
Wait banner :
```

Example

The following example shows the **show platform software configuration access policy** command being issued both before and after a new transport map for SSH are configured. During the configuration, the connection policy and banners are set for a persistent SSH transport map, and the transport map for SSH is enabled.

```
Router# show platform software configuration access policy
The current access-policies

Method : telnet
Rule : wait with interrupt
Shell banner:
Welcome to Diagnostic Mode

Wait banner :
Waiting for IOS Process
```

```
Method : ssh
Rule : wait
Shell banner:
Wait banner :

Method : console
Rule : wait with interrupt
Shell banner:
Wait banner :

Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# transport-map type persistent ssh sshhandler
Router(config-tmap)# connection wait allow interruptible
Router(config-tmap)# banner diagnostic X
Enter TEXT message. End with the character 'X'.
Welcome to Diag Mode
X
Router(config-tmap)# banner wait X
Enter TEXT message. End with the character 'X'.
Waiting for IOS
X
Router(config-tmap)# rsa keypair-name sshkeys
Router(config-tmap)# transport interface gigabitethernet 0
Router(config-tmap)# exit
Router(config)# transport type persistent ssh input sshhandler
Router(config)# exit

Router# show platform software configuration access policy
The current access-policies

Method : telnet
Rule : wait with interrupt
Shell banner:
Welcome to Diagnostic Mode

Wait banner :
Waiting for IOS process


Method : ssh
Rule : wait with interrupt
Shell banner:
Welcome to Diag Mode

Wait banner :
Waiting for IOS


Method : console
Rule : wait with interrupt
Shell banner:
Wait banner :
```

Configuring Auxiliary Port for Modem Connection

Cisco 4000 Series ISR supports connecting a modem to the router auxiliary port for EXEC dial in connectivity. When a modem is connected to the auxiliary port, a remote user can dial in to the router and configure it. To configure a modem on the auxiliary port, perform these steps:

Step 1 Connect the RJ-45 end of the adapter cable to the black AUX port on the router.

Step 2 Use the **show line** command to determine the async interface of the AUX port:

```
Router# show  line

 Tty Typ     Tx/Rx    A Modem  Roty AccO AccI   Uses   Noise  Overruns   Int
*     0 CTY            -   -    -  - -   0   0   0/0    -
      1 AUX   9600/9600 -   -    -  - -   0   0   0/0    -
      2 VTY            -   -    -  - -   0   0   0/0    -
      3 VTY            -   -    -  - -   0   0   0/0    -
      4 VTY            -   -    -  - -   0   0   0/0    -
      5 VTY            -    -   -  - -   0   0   0/0    -
      6 VTY            -    -   -  - -   0   0   0/0    -
```

Step 3 Use the following commands to configure the router AUX line::

```
Router(config)# line 1

Router(config-line)#modem inOut
Router(config-line)#modem autoconfigure type usr_sportster
Router(config-line)#speed 115200  [Speed to be set according to the modem manual]
Router(config-line)#stopbits 1 [Stopbits to be set according to the modem manual]
Router(config-line)#transport input all
Router(config-line)#flowcontrol hardware [flowcontrol to be set according to the modem manual]
Router(config-line)#password cisco
Router(config-line)#login
Router(config-line)#end
Router(config)#enable password lab
```

Step 4 Use the reverse telnet method on the modem to verify the modem connectivity and configuration string:

```
Router(config)#int loopback 0
Router(config-if)#ip add 192.0.2.1 255.255.255.0
Router(config-if)#end
Router#telnet 192.0.2.1 2001
Trying 192.0.2.1, 2001 ... Open

User Access Verification

Password: <enter the password given under line configuration>

at    <<<=== Modem command
OK  <<<=== This OK indicates that the modem is connected successully to the AUX port.
```

Step 5 Use an analog phone to verify that the phone line is active and functions properly. Then, connect the analog phone line to the modem.

Step 6 Initialize an EXEC modem call to the router from another device (PC) to test the modem connection.

Step 7 When the connection is established, the dial in client is prompted for a password. Enter the correct password.

Note: This password should match the one that is configured on the auxiliary port line.

CHAPTER 8

Installing the Software

This chapter includes the following sections:

Overview

Installing software on the router involves installing a consolidated package (bootable image). This consists of a bundle of subpackages (modular software units), with each subpackage controlling a different set of functions.

These are the two main methods to install the software:

- Managing and Configuring a Router to Run Using a Consolidated Package, on page 108—This method allows for individual upgrade of subpackages and generally has reduced boot times compared to the method below. Use this method if you want to individually upgrade a module's software.
- Managing and Configuring a Router to Run Using Individual Packages, on page 113—This a simple method that is similar to a typical Cisco router image installation and management that is supported across Cisco routers.

It is better to upgrade software in a planned period of maintenance when an interruption in service is acceptable. The router needs to be rebooted for a software upgrade to take effect.

ROMMON Images

A ROMMON image is a software package used by ROM Monitor (ROMMON) software on a router. The software package is separate from the consolidated package normally used to boot the router. For more information on ROMMON, see the "ROM Monitor Overview and Basic Procedures" section in the Upgrading Field-Programmable Hardware Devices for Cisco 4000 Series ISRs guide.

An independent ROMMON image (software package) may occasionally be released and the router can be upgraded with the new ROMMON software. For detailed instructions, see the documentation that accompanies the ROMMON image.

Note A new version of the ROMMON image is not necessarily released at the same time as a consolidated package for a router.

Rommon Compatibility Matrix

The following table provides information about Cisco 4000 Series Integrated Services Routers supported in each ROMMON release.

Table 12: Supported ROMMON Releases for Cisco 4000 Series Integrated Service Routers

Platform	16.2(1r)	16.2(2r)	16.4(3r)	16.7(3r)	16.7(4r)	16.7(5r)	16.8(1r)	16.9(1r)	16.12(1r)	16.12(2r)	17.6.1
Cisco 4221 ISR	—	—	Yes	Yes	Yes	Yes	—	Yes	Yes	Yes	Yes
Cisco 4321 ISR	Yes	Yes	Yes	Yes	Yes	Yes	—	Yes	Yes	Yes	Yes
Cisco 4331 ISR	Yes	Yes	Yes	Yes	Yes	Yes	—	Yes	Yes	Yes	Yes
Cisco 4351 ISR	Yes	Yes	Yes	Yes	Yes	Yes	—	Yes	Yes	Yes	Yes
Cisco 4431 ISR	Yes	—	—	—	Yes	Yes	—	—	—	Yes	Yes
Cisco 4451 ISR	Yes	—	—	—	Yes	Yes	—	—	—	Yes	Yes
Cisco 4461 ISR	—	—	—	—	—	—	—	Yes	Yes	Yes	Yes

Note When you upgrade from Cisco IOS XE 3.x to 16.x image, you should first upgrade the rommon release to the 16.7(5r) rommon release. After upgrading to the 16.7(5r) rommon release, based on the IOS XE 16.x image, the rommon release can be auto-upgraded to a later rommon release.

Note The rommon release 16.9(1r) is the first release that supports the Cisco BIOS Protection. After a device is upgraded to the 16.9(1r) rommon release, the rommon release cannot be downgraded to a release earlier than 16.9(1r). All future rommon releases can be downgraded to the 16.9(1r) release. Also, if a platform has a 16.9(1r) or later release installed, an IOS XE 16.9.1 or later release or a SD-WAN 16.11.1 or later release must be used for the upgrade.

Note ROMMON images for IOS XE Release 17.1.x through 17.5.x are aligned with release 16.12(2r).

Note From Cisco IOS XE Release 17.6.1 onwards, the ROMMON image will not be released as a standalone package, and will be packaged with the IOS XE image. 17.6.1 ROMMON will only be used in devices with manufacturing date equal or later than 2535. You can view your device manufacturing date with the CLI command **show license udi**. For example,

```
elixir_p1b_11#show  license udi
 UDI: PID:C1131X-8PWB, SN: FGL2451L5MJ
```

The device manufacturing date in this example is 2451.

Minimum Supported ROMMON Release

The following table provides the minimum supported ROMMON release in Cisco IOS XE 16.x..x releases.

Table 13: Minimum Supported ROMMON Release in Cisco IOS XE 16.x.x Releases

Cisco IOS XE Release	Cisco 4321 ISR	Cisco 4321 ISR	Cisco 4331 ISR	Cisco 4351 ISR	Cisco 4431 ISR	Cisco 4451 ISR	Cisco 4461 ISR
Cisco IOS XE 16.3.x	—	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	—
Cisco IOS XE 16.4.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	—
Cisco IOS XE 16.5.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	—
Cisco IOS XE 16.6.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	—

Cisco IOS XE Release	Cisco 4321 ISR	Cisco 4321 ISR	Cisco 4331 ISR	Cisco 4351 ISR	Cisco 4431 ISR	Cisco 4451 ISR	Cisco 4461 ISR
Cisco IOS XE 16.7.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	—
Cisco IOS XE 16.8.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	—
Cisco IOS XE 16.9.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	16.9(1r)
Cisco IOS XE 16.10.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	16.9(1r)
Cisco IOS XE 16.11.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	16.9(1r)
Cisco IOS XE 16.12.x	16.7(4r)	16.7(3r)	16.7(3r)	16.7(3r)	16.7(4r)	16.7(4r)	16.9(1r)
Cisco IOS XE 17.1.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.2.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.3.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.4.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.5.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.6.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)

Note For devices with manufacturing date equal or later than 2535, the minimum supported ROMMON version is 17.6.1. These devices cannot downgrade to older ROMMON versions.

Recommended ROMMON Release

The following table lists the recommended ROMMON release for the routing platforms in each Cisco IOS XE 16.x.x releases.

Table 14: Recommended ROMMON Release for Cisco IOS XE 16.x.x Releases

Cisco IOS XE Release	Cisco 4321 ISR	Cisco 4321 ISR	Cisco 4331 ISR	Cisco 4351 ISR	Cisco 4431 ISR	Cisco 4451 ISR	Cisco 4461 ISR
Cisco IOS XE 16.3.x	—	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	—
Cisco IOS XE 16.4.x	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	—
Cisco IOS XE 16.5.x	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	—
Cisco IOS XE 16.6.x	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	—
Cisco IOS XE 16.7.x	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	—
Cisco IOS XE 16.8.x	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	16.7(5r)	—
Cisco IOS XE 16.9.x	16.9(1r)	16.9(1r)	16.9(1r)	16.9(1r)	16.12(2r)	16.12(2r)	16.9(1r)
Cisco IOS XE 16.10.x	16.9(1r)	16.9(1r)	16.9(1r)	16.9(1r)	16.12(2r)	16.12(2r)	16.9(1r)
Cisco IOS XE 16.11.x	16.9(1r)	16.9(1r)	16.9(1r)	16.9(1r)	16.12(2r)	16.12(2r)	16.9(1r)
Cisco IOS XE 16.12.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.1.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.2.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.3.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.4.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.5.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)
Cisco IOS XE 17.6.x	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)	16.12(2r)

> **Note** For devices with manufacturing date equal or later than 2535, the minimum supported ROMMON version is 17.6.1. These devices cannot downgrade to older ROMMON versions. For devices with IOS XE 16.12 and preinstalled ROMMON 17.6.1r, the minimum supported ROMMON version is 17.6.1r. Do not downgrade the ROMMON to 16.12(2r); these devices cannot downgrade to older ROMMON versions.

Provisioning Files

This section provides background information about the files and processes used in Managing and Configuring a Router to Run Using Individual Packages, on page 113.

The consolidated package on a router consists of a collection of subpackages and a provisioning file titled `packages.conf`. To run the software, the usual method used is to boot the consolidated package, which is copied into memory, expanded, mounted, and run within memory. The provisioning file's name can be renamed but subpackage file's names cannot be renamed. The provisioning file and subpackage files must be kept in the same directory. The provisioning file does not work properly if any individual subpackage file is contained within a different directory.

> **Note** An exception to this is that if a new or upgraded module firmware package is subsequently installed, it need not be in the same directory as the provisioning file.

Configuring a router to boot, using the provisioning file packages.conf, is beneficial because no changes have to be made to the boot statement after the Cisco IOS XE software is upgraded.

File Systems

The following table provides a list of file systems that can be seen on the Cisco 4000 series routers.

Table 15: Router File Systems

File System	Description
bootflash:	Boot flash memory file system.
flash:	Alias to the boot flash memory file system above.
harddisk:	Hard disk file system (if NIM-SSD, NIM-HDD, or internal mSATA flash device is present in the router). **Note** The internal mSATA flash device is supported only on Cisco ISR4300 Series routers.
cns:	Cisco Networking Services file directory.
nvram:	Router NVRAM. You can copy the startup configuration to NVRAM or from NVRAM.

File System	Description
obfl:	File system for Onboard Failure Logging (OBFL) files.
system:	System memory file system, which includes the running configuration.
tar:	Archive file system.
tmpsys:	Temporary system files file system.
usb0: usb1:	The Universal Serial Bus (USB) flash drive file systems. **Note** The USB flash drive file system is visible only if a USB drive is installed in usb0: or usb1: ports.

Use the **?** help option, or use the **copy** command in command reference guides, if you find a file system that is not listed in the table above.

Autogenerated File Directories and Files

This section discusses the autogenerated files and directories that can be created, and how the files in these directories can be managed.

Table 16: Autogenerated Files

File or Directory	Description
crashinfo files	Crashinfo files may appear in the bootflash: file system. These files provide descriptive information of a crash and may be useful for tuning or troubleshooting purposes. However, the files are not part of router operations, and can be erased without impacting the functioning of the router.
core directory	The storage area for .core files. If this directory is erased, it will automatically regenerate itself at bootup. The .core files in this directory can be erased without impacting any router functionality, but the directory itself should not be erased.
lost+found directory	This directory is created on bootup if a system check is performed. Its appearance is completely normal and does not indicate any issues with the router.
tracelogs directory	The storage area for trace files. Trace files are useful for troubleshooting. If the Cisco IOS process fails, for instance, users or troubleshooting personnel can access trace files using diagnostic mode to gather information related to the Cisco IOS failure. Trace files, however, are not a part of router operations, and can be erased without impacting the router's performance.

Important Notes About Autogenerated Directories

Important information about autogenerated directories include:

- Autogenerated files on the bootflash: directory should not be deleted, renamed, moved, or altered in any way unless directed by Cisco customer support.

> **Note** Altering autogenerating files on the bootflash: may have unpredictable consequences for system performance.

- Crashinfo, core, and trace files can be deleted.

Flash Storage

Subpackages are installed to local media storage, such as flash. For flash storage, use the **dir bootflash:** command to list the file names.

> **Note** Flash storage is required for successful operation of a router.

Configuring the Configuration Register for Autoboot

The configuration register can be used to change router behavior. This includes controlling how the router boots. Set the configuration register to 0x0 to boot into ROM, by using one of the following commands:

- In Cisco IOS configuration mode, use the **config-reg** 0x0 command.
- From the ROMMON prompt, use the **confreg** 0x0 command.

For more information about the configuration register, see Use of the Configuration Register on All Cisco Routers and Configuring a Router to Boot the Consolidated Package via TFTP Using the boot Command: Example, on page 109.

> **Note** Setting the configuration register to 0x2102 will set the router to autoboot the Cisco IOS XE software.

> **Note** The console baud rate is set to 9600 after changing the **confreg** to 0x2102 or 0x0. If you cannot establish a console session after setting **confreg**, or garbage output appears, change the setting on your terminal emulation software to 9600.

Licensing

Cisco Software Licensing

Cisco software licensing consists of processes and components to activate Cisco IOS software feature sets by obtaining and validating Cisco software licenses.

You can enable licensed features and store license files in the bootflash of your router. Licenses pertain to consolidated packages, technology packages, or individual features.

An evaluation license is automatically converted to a Right to Use model after 60 days and this license is valid permanently. The conversion to a permanent license applies only to evaluation licenses. For other features supported on your router, you must purchase a permanent license.

See the "Configuring the Cisco IOS Software Activation Feature" chapter of the Software Activation Configuration Guide, Cisco IOS XE Release 3S.

Consolidated Packages

One of the following two consolidated packages (images) is preinstalled on the router:

- **universalk9**—Contains the **ipbasek9** base package and the **securityk9**, **uck9**, and **appxk9** technology packages.
- **universalk9_npe**—Contains the **ipbasek9** base package and the **securityk9_npe**, **uck9**, and **appxk9** technology packages. This image has limited crypto functionality.

Note The term npe stands for No Payload Encryption.

Note The terms super package and image also refer to a consolidated package.

To obtain software images for the router, go to http://software.cisco.com/download/navigator.html.

An image-based license is used to help bring up all the subsystems that correspond to a license. This license is enforced only at boot time.

Apart from the **universalk9** and **universalk9_npe** images, a Boot ROMMON image is available. For more information, see *ROMMON Images* section.

For more information about identifying digitally signed Cisco software and how to show the digital signature information of an image file, see the "Digitally Signed Cisco Software" section in the Loading and Managing System Images Configuration Guide, Cisco IOS XE Release 3S.

The following examples show how to obtain software authenticity information and internal details of a package:

- *Displaying Digitally Signed Cisco Software Signature Information* section
- *Obtaining the Description of a Module or Consolidated Package* section

Many features within the consolidated package are contained in the **ipbasek9** base package. The license key for the **ipbasek9** package is activated by default.

Technology Packages

Technology packages contain software features within a consolidated package. To use different sets of features, enable the licenses of selected technology packages. You can enable the licenses for any combination of technology packages.

Each technology package has an evaluation license that converts to a Right to Use (RTU) license after 60 days and is then valid permanently.

The following is a list of technology packages:

Note In Cisco 1000 Series Integrated Series Routers, although L2TPv2 sessions comes up without appxk9, you need the appxk9 license for the traffic to go through the sessions. You also need the appxk9 license to apply the QoS policies to the L2TPv2 sessions.

securityk9

The **securityk9** technology package includes all crypto features, including IPsec, SSL/SSH, Firewall, and Secure VPN.

The **securityk9_npe** package (npe = No Payload Encryption) includes all the features in the **securityk9** technology package without the payload-encryption functionality. This is to fulfill export restriction requirements. The **securityk9_npe** package is available only in the **universalk9_npe** image. The difference in features between the **securityk9** package and the **securityk9_npe** package is therefore the set of payload-encryption-enabling features such as IPsec and Secure VPN.

uck9

The Unified Communications technology package is required to enable Cisco Unified Border Element (Cisco UBE) functionality. To use Cisco UBE features, you will require session licenses and a Security technology package to secure the media.

appxk9

The **appxk9** technology package contains Application Experience features, which are similar to the features in the DATA package of the Cisco Integrated Services Routers Generation 2 routers. For more information, see: http://www.cisco.com/c/en/us/products/collateral/cloud-systems-management/software-activation-on-integrated-services-routers-isr/white_paper_c11_556985.html#wp9000791.

There are many features in the **appxk9** package, including MPLS, PfR, L2/L3 VPN, Broadband, and AVC.

Feature Licenses

To use each of the following features, enable a corresponding feature license, as explained in the following sections:

HSECK9

The **HSECK9** license is required for a feature to have full crypto functionality. Without the **HSECK9** license, only 225 secure tunnels and 85 Mbps of crypto bandwidth would be available. The **HSECK9** license allows features in the **securityk9** technology package to use the maximum number of secure tunnels and crypto bandwidth. To enable the **HSECK9** license, purchase the **FL-44-HSEC-K9** license from Cisco.com and install it using the **license install** *license-files* command. For further information on obtaining and installing feature licenses, see Configuring the Cisco IOS Software Activation Feature.

Note The **HSECK9** feature does not have an evaluation license that converts to an RTU license after 60 days; a feature license must be obtained.

If you do not enable the export control functionality, the device does not send the HSECK9 license request to the Smart Licensing server even if the HSECK9 license feature is configured on the device.

Note Starting from IOS XE Fuji 16.8.1, limits for number of tunnels and crypto throughput are enhanced. Without HSEC, the new throughput limit is 250 Mbps each direction and number of tunnels is 1000.

To enable the license for the **HSECK9** feature, the **securityk9** technology package is also required. For more information about the **securityk9** technology package, see securityk9, on page 102.

Performance

The performance feature, which allows for increased throughput, is enabled by the performance license. This feature is part of the **ipbasek9** technology package. To enable the feature, order the performance license (part number FL-44-PERF-K9). The license is displayed as the throughput license.

You can upgrade the throughput of the ESP from 2.5 Gbps to 5 Gbps by activating the right-to-use license and then reloading the router. For more information on the right-to-use license activation, see **Configuring Cisco Right-To-Use License Configuration Guide**. If you want to determine the current throughput level of the ESP, run the show platform hardware throughput level command. The following example shows the output of this command before the performance upgrade license is applied:

To configure the throughput level, perform the following steps and to upgrade the throughput level use the `platform hardware throughput level { 2500000 | 5000000}` command.

1. In the user EXEC configuration mode, enter the `enable` command.
2. Enter `configure terminal` command to enter the global configuration mode.
3. To upgrade the throughput level, enter the `platform hardware throughput level{2500000|5000000}` command.
4. To exit global configuration mode, enter exit.
5. To save the configuration, enter the `copy running-config startup-config` command.
6. To reload the router enter reload. A reload is required to activate the throughput level.

```
show platform hardware throughput level
The current throughput level is 2500000 kb/s
```

To configure the throughput level, perform the following steps and to upgrade the throughput level use the `platform hardware throughput level { 2500000 | 5000000}` command.

1. In the user EXEC configuration mode, enter the `enable` command.
2. Enter `configure terminal` command to enter the global configuration mode.
3. To upgrade the throughput level, enter the `platform hardware throughput level{2500000|5000000}` command.
4. To exit global configuration mode, enter exit.
5. To save the configuration, enter the `copy running-config startup-config` command.
6. To reload the router enter reload. A reload is required to activate the throughput level.

The following example shows how to upgrade the throughput level:

```
Router>enable
Router#configure terminal
Router(config)#platform hardware throughput level 5000000
% The config will take effect on next reboot
Router(config)#exit
Router#copy running-config startup-config
Router#reload
```

Boost Performance Licenses

Cisco Boost performance license allows you to increase the throughput bandwidth.You can enable Boost performance license in the follwoing modes:

Note To use the Boost performance license, the device must be running the Cisco IOS XE software version 16.07.01 or later. Also, the boost license command will not be available if the device is registered in CSSM before the license is added to license CSSM repository. You have to deregister and register back the device from the CSSM to execute the boost license command.

Note When you enable boost license on Cisco 4000 Series ISRs, you cannot configure the virtual-service container for Snort IPS and ISR-WAAS.

Activating Boost Performance License in CSL Mode

To activate the Boost performance license in Cisco Software License (CSL) mode, peform the following steps:

1. Configure the device with the **license install bootflash:xxx** command as shown in this example.

   ```
   Device#license install bootflash:FDO203520HU_20180409020344635 0.lic
   Installing licenses from "bootflash:FDO203520HU_20180409020344635 0.lic"
   Installing...Feature:booster_performance...Successful:Supported
   1/1 licenses were successfully installed
   0/1 licenses were existing licenses
   0/1 licenses were failed to install

   Building configuration...
   ```

```
[OK]
% Throughput boost is configured, it will take effect after reload
```

2. The following message will be displayed in the logs.

```
*Apr  9 07:40:11.674: %LICENSE-6-INSTALL: Feature booster_performance 1.0 was installed
 in this device.
 UDI=ISR4331/K9:FDO203520HU; StoreIndex=2:Primary License Storage
```

3. The **platform hardware throughput level boost** is automatically added to the configuration.

```
Device#show running-config | include throughput

platform hardware throughput level boost
```

4. Save the configuration and reload the device to enable Boost performance license. After the reload, the Boost Performance is activated as shown in this example.

```
Device#show platform hardware throughput level

The current throughput level is unthrottled

Device#show license

<output omitted>

Index 11 Feature: booster_performance
 Period left: Life time
 License Type: Permanent
 License State: Active, In Use
 License Count: Non-Counted
 License Priority: Medium
```

5. To exit global configuration mode, enter exit.

6. To save the configuration, enter the `copy running-config startup-config` command.

Boost Performance License in Smart License Mode

This section describes the processes to activate and deactivate the Boost performance license from the device with two use-cases.

Enable the boost performance license:

- Boot the device in Smart License mode. The boost performance command is not visible without registering in the Smart Portal.
- After successfully registering to the Smart Portal, check the availability of the boost performance licenses in the smart account.
- Use the **platform hardware throughput level boost** command to enable the feature. You need to save the configuration. If a valid license is still available in the smart account, the Boost Performance feature is enabled after the device is reloaded.
- To check for the platform hardware throughput level, use the **show platform hardware throughput level boost** command. If there are not enough licenses, it shows an Out of Compliance (OOC) message, and the throughput level change does not take effect even after the device is reloaded.

Return of license:

- The device is in the smart license mode with **boost performance** command configured.

- Use **show running-config** and the **show license summary** commands to display the boost performance information from the smart account.
- Use the **no platform hardware throughput level boost** command to disable the functionality.

Note The command is removed from the configuration, but the license is released only after the device is reloaded.

The throughput level does not take effect until the device is reloaded.

The license visibility is available till the device is reloaded.

One count of boost performance license is reduced from the usage pool, and one license is returned to its original pool.

Cisco Software License to Smart Licensing

This section describe a use-case when the device is moving from Cisco Software License(CSL) to Smart License when **boost performance license** is on CSL. The boost performance behavior is determined by the availability of the license in its Smart Account with Boost Performance activated in CSL:

To configure the throughput level, perform the following steps and to upgrade the throughput level use the

1. Configure the device with the **platform hardware throughput level boost** command and then use **show running-config** to check if the boost performance license is activated.
2. Use **show license** to verify if boost performance is in use and in a permanent license mode.
3. Enable smart license by `license smart enable` command. After registration in success, the license request is sent to the smart portal for validation. Boost performance is valid if successful, no reload is required. Otherwise the **platform hardware throughput level boost** is unattached from configuration. Boost performance functionality is disabled after reload.
4. During the transition but before the registration, we have to maintain the Evaluation mode for the license if the is existing to avoid an extra reload later.
5. To exit global configuration mode, enter exit.
6. To save the configuration, enter the `copy running-config startup-config` command.
7. To reload the router enter reload. A reload is required to activate the throughput level.

Smart Licensing to Cisco Software Licensing

This section includes these two use-cases that describe what happens during the transition from Smart License to Cisco Software License.

When boost performance is in use:

- Device # **platform hardware throughput level boost**
- Device# **show license** to ensure that Smart License and Boost performance licenses are enabled.
- Check the Smart License Account if the boost performance license is consumed from the corresponding device.
- Remove Smart License

- Device# **no license smart enable**
- Check the availability of the boost performance license, you may decide to retain the boost command.
- No extra reload is required.

When boost performance is not in use:

- Use **no platform hardware throughput level boost** in the show running-configuration.
- Device # **show license** to check if smart license is enabled, but boost performance license is not in the list.
- Check the Smart License Account, the boost performance license is not used from the corresponding device.
- To remove Smart License, use **no license smart enable**
- Check the availability of the **boost permanent license** to add the **boost** keyword.
- Boost Performance is activated and is in-use after reload

Note If there is no permanent license available, then **no boost performance**command and functionality is likely to change.

- **When hybrid Cisco IOS XE Release is in use:**
- When you use the hybrid Cisco IOS XE Release (IOS XE 16.9.x) and want to rollback from Smart license to right-to-use (RTU) license, you must reload the router twice to move the license to the "Active, In-Use" state.
- Device# **configuration terminal**
- To remove Smart License, use **no license smart enable**
- Device# **no license smart enable**
- Device# **exist**
- To remove Smart License, reload the router.
- Device# **configure terminal**
- Enter **yes** to accept the end-user license agreement.
- Device# **exist**
- To move RTU license to In-Use state, reload the router.

LED Indicators

For information on LEDs on the router, see "LED Indicators" in the "Overview" section of the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

For information on LEDs on the SSD Carrier Card NIM, see "Overview of the SSD Carrier Card NIM (NIM-SSD)" in the "Installing and Upgrading Internal Modules and FRUs" section of the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

Related Documentation

For further information on software licenses, see Software Activation on Cisco Integrated Services Routers and Cisco Integrated Service Routers G2.

For further information on obtaining and installing feature licenses, see Configuring the Cisco IOS Software Activation Feature.

How to Install and Upgrade the Software

To install or upgrade the software, use one of the following methods to use the software from a consolidated package or an individual package. Also see the overview section.

- Managing and Configuring a Router to Run Using a Consolidated Package, on page 108
- Managing and Configuring a Router to Run Using Individual Packages, on page 113

Managing and Configuring a Router to Run Using a Consolidated Package

Note Do not use these procedures if you also need to install any optional subpackages or plan to upgrade individual subpackages. See Managing and Configuring a Router to Run Using Individual Packages, on page 113.

- Managing and Configuring a Consolidated Package Using copy and boot Commands, on page 108
- Configuring a Router to Boot the Consolidated Package via TFTP Using the boot Command: Example, on page 109

Managing and Configuring a Consolidated Package Using copy and boot Commands

To upgrade a consolidated package, copy the consolidated package to the **bootflash:** directory on the router using the **copy** command. After making this copy of the consolidated package, configure the router to boot using the consolidated package file.

The following example shows the consolidated package file being copied to the **bootflash:** file system via TFTP. The config register is then set to boot using **boot system** commands, and the **boot system** commands instruct the router to boot using the consolidated package stored in the **bootflash:** file system. The new configuration is then saved using the **copy running-config startup-config** command, and the system is then reloaded to complete the process.

```
Router# dir bootflash:
Directory of bootflash:/
11 drwx 16384 Dec 4 2007 04:32:46 -08:00 lost+found
86401 drwx 4096 Dec 4 2007 06:06:24 -08:00 .ssh
14401 drwx 4096 Dec 4 2007 06:06:36 -08:00 .rollback_timer
28801 drwx 4096 Mar 18 2008 17:31:17 -07:00 .prst_sync
43201 drwx 4096 Dec 4 2007 04:34:45 -08:00 .installer

928862208 bytes total (712273920 bytes free)
```

```
Router# copy tftp: bootflash:
Address or name of remote host []? 172.17.16.81
Source filename []? /auto/tftp-users/user/isr4400-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
Destination filename [isr4400-universalk9.03.10.00.S.153-3.S-ext.SPA.bin]?
Accessing
tftp://172.17.16.81//auto/tftp-users/user/isr4400-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
...
Loading /auto/tftp-users/user/isr4400-universalk9.03.10.00.S.153-3.S-ext.SPA.bin from
172.17.16.81 (via GigabitEthernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!
[OK - 208904396 bytes]
208904396 bytes copied in 330.453 secs (632176 bytes/sec)
Router# dir bootflash:
Directory of bootflash:/
11 drwx 16384 Dec 4 2007 04:32:46 -08:00 lost+found
86401 drwx 4096 Dec 4 2007 06:06:24 -08:00 .ssh
14401 drwx 4096 Dec 4 2007 06:06:36 -08:00 .rollback_timer
28801 drwx 4096 Mar 18 2008 17:31:17 -07:00 .prst_sync
43201 drwx 4096 Dec 4 2007 04:34:45 -08:00 .installer
12 -rw- 208904396 May 28 2008 16:17:34 -07:00
isr4400-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
928862208 bytes total (503156736 bytes free)
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# boot system flash bootflash:isr4400-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
Router(config)# config-reg 0x2102
Router(config)# exit
Router# show run | include boot
boot-start-marker
boot system flash bootflash:isr4400-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
boot-end-marker
Router# copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
Router# reload
```

Configuring a Router to Boot the Consolidated Package via TFTP Using the boot Command: Example

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#boot system tftp://10.81.116.4/rtp-isr4400-54/isr4400.bin
Router(config)#config-register 0x2102
Router(config)#exit
Router# show run | include boot
boot-start-marker
boot system tftp://10.81.116.4/rtp-isr4400-54/isr4400.bin
boot-end-marker
license boot level adventerprise
Router# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Router# reload
Proceed with reload? [confirm]
Sep 13 17:42:54.445 R0/0: %PMAN-5-EXITACTION: Process manager is exiting: process exit with

reload chassis code
```

```
Initializing Hardware ...

System integrity status: c0000600
Failures detected:
Boot FPGA corrupt


Key Sectors:(Primary,GOOD),(Backup,GOOD),(Revocation,GOOD)
Size of Primary = 2288 Backup = 2288 Revocation = 300

ROM:RSA Self Test Passed
ROM:Sha512 Self Test Passed
Self Tests Latency: 58 msec


System Bootstrap, Version 12.2(20120618:163328) [username-ESGROM_20120618_GAMMA 101],
DEVELOPMENT SOFTWARE
Copyright (c) 1994-2014 by cisco Systems, Inc.
Compiled Mon 05/27/2014 12:39:32.05 by username

Current image running: Boot ROM0

Last reset cause: LocalSoft

Cisco ISR 4400 platform with 4194304 Kbytes of main memory


IP_ADDRESS: 172.18.42.119
IP_SUBNET_MASK: 255.255.255.0
DEFAULT_GATEWAY: 172.18.42.1
TFTP_SERVER: 10.81.116.4
TFTP_FILE: rtp-isr4400-54/isr4400.bin
TFTP_MACADDR: a4:4c:11:9d:ad:97
TFTP_VERBOSE: Progress
TFTP_RETRY_COUNT: 18
TFTP_TIMEOUT: 7200
TFTP_CHECKSUM: Yes
ETHER_PORT: 0

ETHER_SPEED_MODE: Auto Detect
link up...
Receiving rtp-isr4400-54/isr4400.bin from 10.81.116.4
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
File reception completed.
Boot image size = 424317088 (0x194a90a0) bytes

ROM:RSA Self Test Passed
ROM:Sha512 Self Test Passed
Self Tests Latency: 58 msec

Package header rev 1 structure detected
Calculating SHA-1 hash...done
validate_package: SHA-1 hash:
calculated 7294dffc:892a6c35:a7a133df:18c032fc:0670b303
expected 7294dffc:892a6c35:a7a133df:18c032fc:0670b303
Signed Header Version Based Image Detected

Using FLASH based Keys of type = PRIMARY KEY STORAGE
Using FLASH based Keys of type = ROLLOVER KEY STORAGE
RSA Signed DEVELOPMENT Image Signature Verification Successful.
Package Load Test Latency : 5116 msec
Image validated
```

```
%IOSXEBOOT-4-BOOT_ACTIVITY_LONG_TIME: (local/local): load_modules took: 2 seconds,
expected max time 2 seconds

Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706



Cisco IOS Software, ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version
15.4(20140527:095327)
[v154_3_s_xe313_throttle-BLD-BLD_V154_3_S_XE313_THROTTLE_LATEST_20140527_070027-ios 156]
Copyright (c) 1986-2014 by Cisco Systems, Inc.
Compiled Tue 27-May-14 21:28 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2014 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.


Warning: the compile-time code checksum does not appear to be present.
cisco ISR4451/K9 (2RU) processor with 1133585K/6147K bytes of memory.
Processor board ID FGL1619100P
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of Compact flash at bootflash:.
7816688K bytes of USB flash at usb0:.


Press RETURN to get started!
```

```
Router>
Router>
Router>enable
Router# show version
Cisco IOS XE Software, Version BLD_V154_3_S_XE313_THROTTLE_LATEST_20140527_070027-ext
Cisco IOS Software, ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version
15.4(20140527:095327)
v154_3_s_xe313_throttle-BLD-BLD_V154_3_S_XE313_THROTTLE_LATEST_20140527_070027-ios 156]

IOS XE Version: BLD_V154_3_S_XE313_THROTTLE_LATEST

Cisco IOS-XE software, Copyright (c) 2005-2014 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

Router uptime is 0 minutes
Uptime for this control processor is 3 minutes
System returned to ROM by reload
System image file is "tftp://10.81.116.4/rtp-isr4400-54/isr4400.bin"
Last reload reason: Reload Command



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

License Level: adventerprise
License Type: EvalRightToUse
--More-- Next reload license Level: adventerprise

cisco ISR4451/K9 (2RU) processor with 1133585K/6147K bytes of memory.
Processor board ID FGL1619100P
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of Compact flash at bootflash:.
7816688K bytes of USB flash at usb0:.

Configuration register is 0x2102
```

Managing and Configuring a Router to Run Using Individual Packages

To choose between running individual packages or a consolidated package, see *Installing the Software - Overview* section.

The following topics are included in this section:

Installing Subpackages from a Consolidated Package

Perform the following procedure to obtain the consolidated package from a TFTP server.

Another variation of this procedure obtains the consolidated package from a USB flash drive. This is described in Installing Subpackages from a Consolidated Package on a Flash Drive.

Before you begin

Copy the consolidated package to the TFTP server.

SUMMARY STEPS

1. **show version**
2. **dir bootflash:**
3. **show platform**
4. **mkdir bootflash:** *URL-to-directory-name*
5. **request platform software package expand file** *URL-to-consolidated-package* **to** *URL-to-directory-name*
6. **reload**
7. **boot** *URL-to-directory-name*/**packages.conf**
8. **show version installed**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	
	Example:
	
	`Router# show version`
	`Cisco IOS Software, IOS-XE Software`
	`(X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental`
	`Version 15.3(20120627:221639) [build_151722 111]`
	`Copyright (c) 1986-2012 by Cisco Systems, Inc.`
	`Compiled Thu 28-Jun-12 15:17 by mcpre`
	`.`
	`.`
	`.`
		Shows the version of software running on the router. This can later be compared with the version of software to be installed.

	Command or Action	Purpose
Step 2	**dir bootflash:** **Example:** `Router# dir bootflash:`	Displays the previous version of software and that a package is present.

Step 3	**show platform**
	Example:
	`Router# show platform`
	`Chassis type: ISR4451/K9`
		Displays the inventory.

Step 4	**mkdir bootflash:** *URL-to-directory-name* **Example:** `Router# mkdir bootflash:mydir`	Creates a directory to save the expanded software image. You can use the same name as the image to name the directory.
Step 5	**request platform software package expand file** *URL-to-consolidated-package* **to** *URL-to-directory-name* **Example:** `Router# request platform software package expand file bootflash:isr4400-universalk9-NIM.bin to bootflash:mydir`	Expands the software image from the TFTP server (*URL-to-consolidated-package*) into the directory used to save the image (*URL-to-directory-name*), which was created in Step 4.

Step 6	**reload**
	Example:
	`Router# reload`
	`rommon >`
		Enables ROMMON mode, which allows the software in the consolidated file to be activated.

Step 7	**boot** *URL-to-directory-name*/**packages.conf** **Example:** `rommon 1 > boot bootflash:mydir/packages.conf`	Boots the consolidated package, by specifying the path and name of the provisioning file: packages.conf.

Step 8	**show version installed**
	Example:
	`Router# show version installed`
	`Package: Provisioning File, version: n/a, status: active`
		Displays the version of the newly installed software.

Examples

The initial part of the example shows the consolidated package, isr4400-universalk9.164422SSA.bin, being copied to the TFTP server. This is a prerequisite step. The remaining part of the example shows the consolidated file, packages.conf, being booted.

```
Router# copy tftp:isr4400/isr4400-universalk9.164422SSA.bin bootflash:
Address or name of remote host []? 192.0.2.1
Destination filename [isr4400-universalk9.164422SSA.bin]?
Accessing tftp://192.0.2.1/isr4400/isr4400-universalk9.164422SSA.bin...
Loading isr4400/isr4400-universalk9.164422SSA.bin from 192.0.2.1 (via GigabitEthernet0):
!!!!!!!!
```

```
[OK - 410506248 bytes]

410506248 bytes copied in 338.556 secs (1212521 bytes/sec)


Router# show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version

15.3(20120627:221639) [build_151722 111]
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Thu 28-Jun-12 15:17 by mcpre

IOS XE Version: 2012-06-28_15.31_mcpre

Cisco IOS-XE software, Copyright (c) 2005-2012 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

Router uptime is 0 minutes
Uptime for this control processor is 3 minutes
System returned to ROM by reload
System image file is "tftp:isr4400/isr4400.bin"
Last reload reason: Reload Command

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

License Level: adventerprise
License Type: EvalRightToUse
Next reload license Level: adventerprise
cisco ISR4451/K9 (2RU) processor with 1136676K/6147K bytes of memory.
Processor board ID FGL161611AB
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of Compact flash at bootflash:.

Configuration register is 0x8000

Router# dir bootflash:
Directory of bootflash:/

11 drwx 16384 May 3 2012 19:58:37 +00:00 lost+found
```

```
178465 drwx 4096 Jun 6 2012 15:20:20 +00:00 core
584065 drwx 4096 Jul 13 2012 19:19:00 +00:00 .prst_sync
405601 drwx 4096 May 3 2012 19:59:30 +00:00 .rollback_timer
113569 drwx 40960 Jul 13 2012 19:19:32 +00:00 tracelogs
64897 drwx 4096 May 3 2012 19:59:42 +00:00 .installer
13 -rw- 1305 May 7 2012 17:43:42 +00:00 startup-config
14 -rw- 1305 May 7 2012 17:43:55 +00:00 running-config
15 -r-- 1541 Jun 4 2012 18:32:41 +00:00 debug.conf
16 -rw- 1252 May 22 2012 19:58:39 +00:00 running-config-20120522
519169 drwx 4096 Jun 4 2012 15:29:01 +00:00 vman_fdb

7451738112 bytes total (7067635712 bytes free)



Router# show platform
Chassis type: ISR4451/K9

Slot         Type                State                 Insert time (ago)
---------    ------------------- --------------------- -----------------
0            ISR4451/K9          ok                    15:57:33
 0/0         ISR4451-6X1GE       ok                    15:55:24
1            ISR4451/K9          ok                    15:57:33
 1/0         SM-1T3/E3           ok                    15:55:24
2            ISR4451/K9          ok                    15:57:33
 2/0         SM-1T3/E3           ok                    15:55:24
R0           ISR4451/K9          ok, active            15:57:33
F0           ISR4451-FP          ok, active            15:57:33
P0           Unknown             ps, fail              never
P1           XXX-XXXX-XX         ok                    15:56:58
P2           ACS-4450-FANASSY    ok                    15:56:58

Slot         CPLD Version        Firmware Version
---------    ------------------- ---------------------------------------
0            12090323            15.3(01r)S [ciscouser-ISRRO...
1            12090323            15.3(01r)S [ciscouser-ISRRO...
2            12090323            15.3(01r)S [ciscouser-ISRRO...
R0           12090323            15.3(01r)S [ciscouser-ISRRO...
F0           12090323            15.3(01r)S [ciscouser-ISRRO...

Router# mkdir bootflash:isr4400-universalk9.dir1
Create directory filename [isr4400-universalk9.dir1]?
Created dir bootflash:/isr4400-universalk9.dir1
Router# request platform software package expand file bootflash:isr4400-universalk9.NIM.bin

to bootflash:isr4400-universalk9.dir1
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.

Router# reload
Proceed with reload? [confirm]

*Jul 13 19:39:06.354: %SYS-5-RELOAD: Reload requested by console.Reload Reason: Reload
Command.

rommon 1 > boot bootflash:isr4400-universalk9.dir1/packages.conf

File size is 0x00002836
Located isr4400-universalk9.dir1/packages.conf
Image size 10294 inode num 324484, bks cnt 3 blk size 8*512
#
File is comprised of 1 fragments (33%)
```

```
is_valid_sha1hash: SHA-1 hash:
calculated 62f6235a:fc98eb3a:85ce183e:834f1cb3:8a1f71d1
expected 62f6235a:fc98eb3a:85ce183e:834f1cb3:8a1f71d1
File size is 0x04b3dc00
Located isr4400-universalk9.dir1/isr4400-mono-universalk9-build_164422SSA.pkg
Image size 78896128 inode num 324491, bks cnt 19262 blk size 8*512
###########################################################################################
File is comprised of 21 fragments (0%)
.....
```

```
Router# show version installed
Package: Provisioning File, version: n/a, status: active
File: bootflash:isr4400-universalk9.dir1/packages.conf, on: RP0
Built: n/a, by: n/a
File SHA1 checksum: ad09affd3f8820f4844f27acc1add502e0b8f459

Package: rpbase, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-mono-universalk9-build_164422SSA.pkg, on:
 RP0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 5e95c9cbc4eaf5a4a5a1ac846ee2d0f41d1a026b

Package: firmware_attributes, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_attributes_164422SSA.pkg, on:
RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 71614f2d9cbe7f96d3c6e99b67d514bd108c6c99

Package: firmware_dsp_sp2700, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_dsp_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 8334565edf7843fe246783b1d5c6ed933d96d79e
Package: firmware_fpge, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_fpge_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: eb72900ab32c1c50652888ff486cf370ac901dd7

Package: firmware_sm_1t3e3, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_sm_1t3e3_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 803005f15d8ea71ab088647e2766727ac2269871

Package: rpcontrol, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-mono-universalk9_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 980fd58fe581e9346c44417b451d1c09ebb640c2

Package: rpios-universalk9, version: dir1, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-mono-universalk9_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.23, by: mcpre
File SHA1 checksum: 27084f7e30a1d69d45a33e05d1b00345040799fb
Package: rpaccess, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-mono-universalk9_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 0119802deda2da91c38473c47a998fb3ed423448

Package: firmware_attributes, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_attributes_164422SSA.pkg, on:
RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 71614f2d9cbe7f96d3c6e99b67d514bd108c6c99
```

```
Package: firmware_dsp_sp2700, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_dsp_164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 8334565edf7843fe246783b1d5c6ed933d96d79e

Package: firmware_fpge, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_fpge-BLD-BLD_MCP_DEV_LATEST_20120710
_164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: eb72900ab32c1c50652888ff486cf370ac901dd7

Package: firmware_sm_1t3e3, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_sm_1t3e3-BLD-BLD_MCP_DEV_LATEST_
20120710_164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 803005f15d8ea71ab088647e2766727ac2269871

Package: rpcontrol, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-rpcontrol-BLD-BLD_MCP_DEV_LATEST_20120710_
164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 980fd58fe581e9346c44417b451d1c09ebb640c2

Package: rpios-universalk9, version: 2012-07-10_16.23_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-rpios-universalk9-BLD-BLD_MCP_DEV_LATEST_
20120710_164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.23, by: mcpre
File SHA1 checksum: 27084f7e30a1d69d45a33e05d1b00345040799fb

Package: rpaccess, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-rpaccess-BLD-BLD_MCP_DEV_LATEST_20120710_
164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 0119802deda2da91c38473c47a998fb3ed423448

Package: rpbase, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-rpbase-BLD-BLD_MCP_DEV_LATEST_20120710_
164422SSA.pkg, on: RP1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 5e95c9cbc4eaf5a4a5a1ac846ee2d0f41d1a026b

Package: firmware_attributes, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_attributes-BLD-BLD_MCP_DEV_LATEST
_20120710_164422SSA.pkg, on: RP1/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 71614f2d9cbe7f96d3c6e99b67d514bd108c6c99

Package: firmware_dsp_sp2700, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_dsp_sp2700-BLD-BLD_MCP_DEV_LATEST_
20120710_164422SSA.pkg, on: RP1/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 8334565edf7843fe246783b1d5c6ed933d96d79e

Package: firmware_fpge, version: 2012-07-10_16.22_mcpre, status: n/a
```

Installing Subpackages from a Consolidated Package on a Flash Drive

The steps for installing subpackages from a consolidated package on a USB flash drive are similar to those described in Installing Subpackages from a Consolidated Pacakage section .

Step 1 **show version**

Step 2 **dir usb***n***:**

Step 3 **show platform**

Step 4 **mkdir bootflash:***URL-to-directory-name*

Step 5 **request platform software package expand fileusb*n*:** ***package-name to URL-to-directory-name***

Step 6 **reload**

Step 7 **boot** *URL-to-directory-name/***packages.conf**

Step 8 **show version installed**

How to Install and Upgrade the Software for Cisco IOS XE Denali Release 16.3

To install or upgrade the software, use one of the following methods to use the software from a consolidated package or an individual package. Also see *Overview* section.

- *Managing and Configuring a Router to Run Using a Consolidated Package* section
- *Managing and Configuring a Router to Run Using Individual Packages* section
- *Configuring a Router to Boot the Consolidated Package via TFTP Using the boot Command: Example* section
- *Upgrading to Cisco IOS XE Denali Release 16.3* section

Upgrading to Cisco IOS XE Denali Release 16.3

Upgrading the device to Cisco IOS XE Denali Release 16.3 for the first time uses the same procedures as specified in the earlier section. In addition, Cisco IOS XE Denali Release 16.3 requires a minimum ROMMON version. When the device boots up with Cisco IOS XE Denali image for the first time, the device checks the installed version of the ROMMON, and upgrades if the system is running an older version. During the upgrade, do not power cycle the device. The system automatically power cycles the device after the new ROMMON is installed. After the installation, the system will boot up with the Cisco IOS XE image as normal.

Note When the device boots up for first time and if the device requires an upgrade, the entire boot process may take several minutes. This process will be longer than a normal boot due to the ROMMON upgrade.

The following example illustrates the boot process of a consolidated package:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#boot system tftp://10.81.116.4/rtp-isr4400-54/isr4400.bin
Router(config)#config-register 0x2102
Router(config)#exit
Router# show run | include boot
boot-start-marker
boot system tftp://10.81.116.4/rtp-isr4400-54/isr4400.bin
boot-end-marker
license boot level adventerprise
Router# copy running-config startup-config
Destination filename [startup-config]?
```

```
Building configuration...
[OK]
Router# reload
Proceed with reload? [confirm]
Sep 13 17:42:54.445 R0/0: %PMAN-5-EXITACTION: Process manager is exiting: process exit with

reload chassis code


Initializing Hardware ...

System integrity status: c0000600


Key Sectors:(Primary,GOOD),(Backup,GOOD),(Revocation,GOOD)
Size of Primary = 2288 Backup = 2288 Revocation = 300

ROM:RSA Self Test Passed
ROM:Sha512 Self Test Passed
Self Tests Latency: 58 msec


System Bootstrap, Version 12.2(20120618:163328) [username-ESGROM_20120618_GAMMA 101],
DEVELOPMENT SOFTWARE
Copyright (c) 1994-2014 by cisco Systems, Inc.
Compiled Mon 05/27/2014 12:39:32.05 by username

Current image running: Boot ROM0

Last reset cause: LocalSoft

Cisco ISR 4400 platform with 4194304 Kbytes of main memory


IP_ADDRESS: 172.18.42.119
IP_SUBNET_MASK: 255.255.255.0
DEFAULT_GATEWAY: 172.18.42.1
TFTP_SERVER: 10.81.116.4
TFTP_FILE: rtp-isr4400-54/isr4400.bin
TFTP_MACADDR: a4:4c:11:9d:ad:97
TFTP_VERBOSE: Progress
TFTP_RETRY_COUNT: 18
TFTP_TIMEOUT: 7200
TFTP_CHECKSUM: Yes
ETHER_PORT: 0

ETHER_SPEED_MODE: Auto Detect
link up...
Receiving rtp-isr4400-54/isr4400.bin from 10.81.116.4
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
File reception completed.
Boot image size = 504063931 (0x1e0b67bb) bytes

ROM:RSA Self Test Passed
ROM:Sha512 Self Test Passed
Self Tests Latency: 58 msec

Package header rev 1 structure detected
Calculating SHA-1 hash...done
validate_package: SHA-1 hash:
calculated 7294dffc:892a6c35:a7a133df:18c032fc:0670b303
expected 7294dffc:892a6c35:a7a133df:18c032fc:0670b303
Signed Header Version Based Image Detected
```

```
Using FLASH based Keys of type = PRIMARY KEY STORAGE
Using FLASH based Keys of type = ROLLOVER KEY STORAGE
RSA Signed DEVELOPMENT Image Signature Verification Successful.
Package Load Test Latency : 5116 msec
Image validated


Detected old ROMMON version 12.2(20150910:184432), upgrade required
Upgrading to newer ROMMON version required by this version of IOS-XE, do not power cycle
the system.  A reboot will automatically occur for the new ROMMON to take effect.
selected : 1
Booted : 1
Reset Reason: 1

Info: Upgrading entire flash from the rommon package
Switching to ROM 0
Upgrade image MD5 signature is b702a0a59a46a20a4924f9b17b8f0887
Upgrade image MD5 signature verification is b702a0a59a46a20a4924f9b17b8f0887
Switching back to ROM 1
ROMMON upgrade complete.

To make the new ROMMON permanent, you must restart the RP.
ROMMON upgrade successful.  Rebooting for upgrade to take effect.


Initializing Hardware ...

System integrity status: 00300610
Key Sectors:(Primary,GOOD),(Backup,GOOD),(Revocation,GOOD)
Size of Primary = 2288 Backup = 2288 Revocation = 300

ROM:RSA Self Test Passed

Expected hash:
ddaf35a193617abacc417349ae204131
12e6fa4e89a97ea20a9eeee64b55d39a
2192992a274fc1a836ba3c23a3feebbd
454d4423643ce80e2a9ac94fa54ca49f

Obtained hash:
ddaf35a193617abacc417349ae204131
12e6fa4e89a97ea20a9eeee64b55d39a
2192992a274fc1a836ba3c23a3feebbd
454d4423643ce80e2a9ac94fa54ca49f
ROM:Sha512 Self Test Passed
Self Tests Latency: 418 msec
Rom image verified correctly


System Bootstrap, Version 12.2(20120618:163328) [username-ESGROM_20120618_GAMMA 101],
DEVELOPMENT SOFTWARE
Copyright (c) 1994-2014 by cisco Systems, Inc.
Compiled Mon 05/27/2014 12:39:32.05 by username

CPLD Version: 33 (MM/DD/YY): 06/23/14 Cisco ISR4351/K9 Slot:0

Current image running: Boot ROM1

Last reset cause: ResetRequest
Reading confreg 0x2102

Reading monitor variables from NVRAM
Enabling interrupts...done
```

```
Checking for PCIe device presence...done
Cisco ISR4351/K9 platform with 16777216 Kbytes of main memory

autoboot entry: NVRAM VALUES: bootconf: 0x0, autobootstate: 0
autobootcount: 0, autobootsptr: 0x0
Rommon upgrade requested
Flash upgrade reset 0 in progress
.......
Initializing Hardware ...

Checking for PCIe device presence...done
Reading confreg 2102
System integrity status: 0x300610
Key Sectors:(Primary, GOOD),(Backup,GOOD),(Revocation,GOOD)
Size of Primary = 2288 Backup = 2288 Revocation = 288
RSA Self Test Passed

Expected hash:
DDAF35A193617ABACC417349AE204131
12E6FA4E89A97EA20A9EEEE64B55D39A
2192992A274FC1A836BA3C23A3FEEBBD
454D4423643CE80E2A9AC94FA54CA49F

Obtained hash:
DDAF35A193617ABACC417349AE204131
12E6FA4E89A97EA20A9EEEE64B55D39A
2192992A274FC1A836BA3C23A3FEEBBD
454D4423643CE80E2A9AC94FA54CA49F
Sha512 Self Test Passed
Rom image verified correctly


System Bootstrap, Version 16.2(1r), RELEASE SOFTWARE
Copyright (c) 1994-2016  by cisco Systems, Inc.


Current image running: *Upgrade in progress* Boot ROM0

Last reset cause: BootRomUpgrade
ISR4351/K9 platform with 16777216 Kbytes of main memory

Cisco ISR 4400 platform with 4194304 Kbytes of main memory


IP_ADDRESS: 172.18.42.119
IP_SUBNET_MASK: 255.255.255.0
DEFAULT_GATEWAY: 172.18.42.1
TFTP_SERVER: 10.81.116.4
TFTP_FILE: rtp-isr4400-54/isr4400.bin
TFTP_MACADDR: a4:4c:11:9d:ad:97
TFTP_VERBOSE: Progress
TFTP_RETRY_COUNT: 18
TFTP_TIMEOUT: 7200
TFTP_CHECKSUM: Yes
ETHER_PORT: 0

ETHER_SPEED_MODE: Auto Detect
link up...
Receiving rtp-isr4400-54/isr4400.bin from 10.81.116.4
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
File reception completed.
Boot image size = 504063931 (0x1e0b67bb) bytes
```

```
Image Base is: 0x56834018
Image Size is: 0x1E089706
Package header rev 1 structure detected
Package type:30000, flags:0x0
IsoSize = 503874534
Parsing package TLV info:
000: 000000090000001D4B45595F544C565F -         KEY_TLV_
010: 5041434B4147455F434F4D5041544942 - PACKAGE_COMPATIB
020: 494C495459000000000000090000000B - ILITY
030: 4652555F52505F545950450000000009 - FRU_RP_TYPE
040: 000000184B45595F544C565F5041434B -     KEY_TLV_PACK
050: 4147455F424F4F54415243480000009 - AGE_BOOTARCH
060: 0000000E415243485F693638365F5459 -     ARCH_i686_TY
070: 50450000000000009000000144B45595F - PE         KEY_
080: 544C565F424F4152445F434F4D504154 - TLV_BOARD_COMPAT
090: 00000009000000012424F4152445F6973 -        BOARD_is
0A0: 72343330305F54595045000000000009 - r4300_TYPE
0B0: 000000184B45595F544C565F43525950 -     KEY_TLV_CRYP
0C0: 544F5F4B4559535452494E4700000009 - TO_KEYSTRING

TLV: T=9, L=29, V=KEY_TLV_PACKAGE_COMPATIBILITY
TLV: T=9, L=11, V=FRU_RP_TYPE
TLV: T=9, L=24, V=KEY_TLV_PACKAGE_BOOTARCH
TLV: T=9, L=14, V=ARCH_i686_TYPE
TLV: T=9, L=20, V=KEY_TLV_BOARD_COMPAT
TLV: T=9, L=18, V=BOARD_isr4300_TYPE
TLV: T=9, L=24, V=KEY_TLV_CRYPTO_KEYSTRING
TLV: T=9, L=10, V=EnCrYpTiOn
TLV: T=9, L=11, V=CW_BEGIN=$$
TLV: T=9, L=19, V=CW_FAMILY=$isr4300$
TLV: T=9, L=59, V=CW_IMAGE=$isr4300-universalk9.2016-06-29_23.31_paj.SSA.bin$
TLV: T=9, L=19, V=CW_VERSION=$16.3.1$
TLV: T=9, L=52, V=CW_DESCRIPTION=$Cisco IOS Software, IOS-XE Software$
TLV: T=9, L=9, V=CW_END=$$
Found DIGISIGN TLV type 12 length = 392
RSA Self Test Passed

Expected hash:
DDAF35A193617ABACC417349AE204131
12E6FA4E89A97EA20A9EEEE64B55D39A
2192992A274FC1A836BA3C23A3FEEBBD
454D4423643CE80E2A9AC94FA54CA49F

Obtained hash:
DDAF35A193617ABACC417349AE204131
12E6FA4E89A97EA20A9EEEE64B55D39A
2192992A274FC1A836BA3C23A3FEEBBD
454D4423643CE80E2A9AC94FA54CA49F
Sha512 Self Test Passed
Found package arch type ARCH_i686_TYPE
Found package FRU type FRU_RP_TYPE
Calculating SHA-1 hash...Validate package: SHA-1 hash:
 calculated 8B082C48:35C23C9E:8A091441:D6FACEE6:B5111533
 expected   8B082C48:35C23C9E:8A091441:D6FACEE6:B5111533

Image validated

Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
```

```
Software clause at DFARS sec. 252.227-7013.

cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706



Cisco IOS Software, ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version
16.3(20160527:095327)
[v163_throttle]
Copyright (c) 1986-2016 by Cisco Systems, Inc.
Compiled Tue 27-May-16 21:28 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2016 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.


Warning: the compile-time code checksum does not appear to be present.
cisco ISR4451/K9 (2RU) processor with 1133585K/6147K bytes of memory.
Processor board ID FGL1619100P
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of Compact flash at bootflash:.
7816688K bytes of USB flash at usb0:.


Press RETURN to get started!
```

Installing a Firmware Subpackage

Before you begin

Obtain a consolidated package that contains your required firmware package and expand the package. (See Managing and Configuring a Router to Run Using Individual Packages, on page 113.) Make a note of the location and name of the firmware package and use this information in the steps below for *URL-to-package-name*.

You can install a firmware subpackage if the router has been configured using, for example, Managing and Configuring a Router to Run Using Individual Packages, on page 113.

Firmware subpackages are not released individually. You can select a firmware package from within a consolidated package after expanding the consolidated package. The firmware package can then be installed as shown in the procedure below.

Note Read the Release Notes document pertaining to the consolidated package to verify that the firmware within the consolidated package is compatible with the version of Cisco IOS XE software that is currently installed on a router.

SUMMARY STEPS

1. **show version**
2. **dir bootflash:**
3. **show platform**
4. **mkdir bootflash:** *URL-to-directory-name*
5. **request platform software package expand file** *URL-to-consolidated-package* **to** *URL-to-directory-name*
6. **reload**
7. **boot** *URL-to-directory-name* **/packages.conf**
8. **show version installed**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show version**
	
	Example:
	`Router# show version`
	`Cisco IOS Software, IOS-XE Software`
	`(X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental`
	`Version 15.3(20120627:221639) [build_151722 111]`
	`Copyright (c) 1986-2012 by Cisco Systems, Inc.`
	`Compiled Thu 28-Jun-12 15:17 by mcpre`
	`.`
	`.`
	`.`
		Shows the version of software running on the router. This can later be compared with the version of software to be installed.

Step 2	**dir bootflash:** **Example:** `Router# dir bootflash:`	Displays the previous version of software and that a package is present.

Step 3	**show platform**
	
	Example:
	`Router# show platform`
	`Chassis type: ISR4451/K9`
		Checks the inventory.
		
		Also see the example in Installing Subpackages from a Consolidated Package section.

	Command or Action	Purpose
Step 4	**mkdir bootflash:** *URL-to-directory-name* **Example:** `Router# mkdir bootflash:mydir`	Creates a directory to save the expanded software image. You can use the same name as the image to name the directory.
Step 5	**request platform software package expand file** *URL-to-consolidated-package* **to** *URL-to-directory-name* **Example:** `Router# request platform software package expand file bootflash:isr4400-universalk9-NIM.bin to bootflash:mydir`	Expands the software image from the TFTP server (*URL-to-consolidated-package*) into the directory used to save the image (*URL-to-directory-name*), which was created in the Step 4.

Step 6	**reload**
	Example:
	`Router# reload`
	`rommon >`
		Enables ROMMON mode, which allows the software in the consolidated file to be activated.

Step 7	**boot** *URL-to-directory-name* **/packages.conf** **Example:** `rommon 1 > boot bootflash:mydir/packages.conf`	Boots the consolidated package by specifying the path and name of the provisioning file: packages.conf.

Step 8	**show version installed**
	Example:
	`Router# show version installed`
	`Package: Provisioning File, version: n/a, status: active`
		Displays the version of the newly installed software.

Examples

The initial part of the following example shows the consolidated package, isr4400-universalk9.164422SSA.bin, being copied to the TFTP server. This is a prerequisite step. The remaining part of the example shows the consolidated file, packages.conf, being booted.

```
Router# tftp:isr4400/isr4400-universalk9.164422SSA.bin bootflash:
Address or name of remote host []? 192.0.2.1
Destination filename [isr4400-universalk9.164422SSA.bin]?
Accessing tftp://192.0.2.1/isr4400/isr4400-universalk9.164422SSA.bin...
Loading isr4400/isr4400-universalk9.164422SSA.bin from 192.0.2.1 (via GigabitEthernet0):
!!!!!!!!
[OK - 410506248 bytes]

410506248 bytes copied in 338.556 secs (1212521 bytes/sec)


Router# show version
Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version

15.3(20120627:221639) [build_151722 111]
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Thu 28-Jun-12 15:17 by mcpre
```

```
IOS XE Version: 2012-06-28_15.31_mcpre

Cisco IOS-XE software, Copyright (c) 2005-2012 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

Router uptime is 0 minutes
Uptime for this control processor is 3 minutes
System returned to ROM by reload
System image file is "tftp:isr4400/isr4400.bin"
Last reload reason: Reload Command

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

License Level: adventerprise
License Type: EvalRightToUse
Next reload license Level: adventerprise
cisco ISR4451/K9 (2RU) processor with 1136676K/6147K bytes of memory.
Processor board ID FGL161611AB
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of Compact flash at bootflash:.

Configuration register is 0x8000

Router# dir bootflash:
Directory of bootflash:/

11 drwx 16384 May 3 2012 19:58:37 +00:00 lost+found
178465 drwx 4096 Jun 6 2012 15:20:20 +00:00 core
584065 drwx 4096 Jul 13 2012 19:19:00 +00:00 .prst_sync
405601 drwx 4096 May 3 2012 19:59:30 +00:00 .rollback_timer
113569 drwx 40960 Jul 13 2012 19:19:32 +00:00 tracelogs
64897 drwx 4096 May 3 2012 19:59:42 +00:00 .installer
13 -rw- 1305 May 7 2012 17:43:42 +00:00 startup-config
14 -rw- 1305 May 7 2012 17:43:55 +00:00 running-config
15 -r-- 1541 Jun 4 2012 18:32:41 +00:00 debug.conf
16 -rw- 1252 May 22 2012 19:58:39 +00:00 running-config-20120522
519169 drwx 4096 Jun 4 2012 15:29:01 +00:00 vman_fdb

7451738112 bytes total (7067635712 bytes free)
```

```
Router# show platform
Chassis type: ISR4451/K9

Slot Type State Insert time (ago)
--------- ------------------- --------------------- -----------------
0 ISR4451/K9 ok 15:57:33
0/0 ISR4451-6X1GE ok 15:55:24
1 ISR4451/K9 ok 15:57:33
1/0 SM-1T3/E3 ok 15:55:24
2 ISR4451/K9 ok 15:57:33
2/0 SM-1T3/E3 ok 15:55:24
R0 ISR4451/K9 ok, active 15:57:33
F0 ISR4451-FP ok, active 15:57:33
P0 Unknown ps, fail never
P1 XXX-XXXX-XX ok 15:56:58
P2 ACS-4450-FANASSY ok 15:56:58

Slot CPLD Version Firmware Version
--------- ------------------- ---------------------------------------
0 12090323 15.3(01r)S [ciscouser-ISRRO...
1 12090323 15.3(01r)S [ciscouser-ISRRO...
2 12090323 15.3(01r)S [ciscouser-ISRRO...
R0 12090323 15.3(01r)S [ciscouser-ISRRO...
F0 12090323 15.3(01r)S [ciscouser-ISRRO...

Router# mkdir bootflash:isr4400-universalk9.dir1
Create directory filename [isr4400-universalk9.dir1]?
Created dir bootflash:/isr4400-universalk9.dir1
Router# request platform software package expand file bootflash:isr4400-universalk9.NIM.bin
 to
 bootflash:isr4400-universalk9.dir1
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.

Router# reload
Proceed with reload? [confirm]

*Jul 13 19:39:06.354: %SYS-5-RELOAD: Reload requested by console. Reload Reason: Reload
Command.

rommon 1 > boot bootflash:isr4400-universalk9.dir1/packages.conf

File size is 0x00002836
Located isr4400-universalk9.dir1/packages.conf
Image size 10294 inode num 324484, bks cnt 3 blk size 8*512
#
File is comprised of 1 fragments (33%)

is_valid_sha1hash: SHA-1 hash:
calculated 62f6235a:fc98eb3a:85ce183e:834f1cb3:8a1f71d1
expected 62f6235a:fc98eb3a:85ce183e:834f1cb3:8a1f71d1
File size is 0x04b3dc00
Located isr4400-universalk9.dir1/isr4400-mono-universalk9-build_164422SSA.pkg
Image size 78896128 inode num 324491, bks cnt 19262 blk size 8*512
###########################################################################################
File is comprised of 21 fragments (0%)
.....
```

```
Router# show version installed
Package: Provisioning File, version: n/a, status: active
File: bootflash:isr4400-universalk9.dir1/packages.conf, on: RP0
Built: n/a, by: n/a
File SHA1 checksum: ad09affd3f8820f4844f27acc1add502e0b8f459

Package: rpbase, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-mono-universalk9-build_164422SSA.pkg, on:
 RP0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 5e95c9cbc4eaf5a4a5a1ac846ee2d0f41d1a026b

Package: firmware_attributes, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_attributes_164422SSA.pkg, on:
RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 71614f2d9cbe7f96d3c6e99b67d514bd108c6c99

Package: firmware_dsp_sp2700, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_dsp_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 8334565edf7843fe246783b1d5c6ed933d96d79e
Package: firmware_fpge, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_fpge_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: eb72900ab32c1c50652888ff486cf370ac901dd7

Package: firmware_sm_1t3e3, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_sm_1t3e3_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 803005f15d8ea71ab088647e2766727ac2269871

Package: rpcontrol, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-mono-universalk9_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 980fd58fe581e9346c44417b451d1c09ebb640c2

Package: rpios-universalk9, version: dir1, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-mono-universalk9_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.23, by: mcpre
File SHA1 checksum: 27084f7e30a1d69d45a33e05d1b00345040799fb
Package: rpaccess, version: 2012-07-10_16.22_mcpre, status: active
File: bootflash:isr4400-universalk9.dir1/isr4400-mono-universalk9_164422SSA.pkg, on: RP0/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 0119802deda2da91c38473c47a998fb3ed423448

Package: firmware_attributes, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_attributes_164422SSA.pkg, on:
RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 71614f2d9cbe7f96d3c6e99b67d514bd108c6c99

Package: firmware_dsp_sp2700, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_dsp_164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 8334565edf7843fe246783b1d5c6ed933d96d79e

Package: firmware_fpge, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_fpge-BLD-BLD_MCP_DEV_LATEST_
20120710_164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: eb72900ab32c1c50652888ff486cf370ac901dd7
```

```
Package: firmware_sm_1t3e3, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_sm_1t3e3-BLD-BLD_MCP_DEV_LATEST_
20120710_164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 803005f15d8ea71ab088647e2766727ac2269871

Package: rpcontrol, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-rpcontrol-BLD-BLD_MCP_DEV_LATEST_20120710_
164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 980fd58fe581e9346c44417b451d1c09ebb640c2

Package: rpios-universalk9, version: 2012-07-10_16.23_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-rpios-universalk9-BLD-BLD_MCP_DEV_LATEST_
20120710_164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.23, by: mcpre
File SHA1 checksum: 27084f7e30a1d69d45a33e05d1b00345040799fb

Package: rpaccess, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-rpaccess-BLD-BLD_MCP_DEV_LATEST_20120710_
164422SSA.pkg, on: RP0/1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 0119802deda2da91c38473c47a998fb3ed423448

Package: rpbase, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-rpbase-BLD-BLD_MCP_DEV_LATEST_20120710_
164422SSA.pkg, on: RP1
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 5e95c9cbc4eaf5a4a5a1ac846ee2d0f41d1a026b

Package: firmware_attributes, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_attributes-BLD-BLD_MCP_DEV_LATEST_
20120710_164422SSA.pkg, on: RP1/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 71614f2d9cbe7f96d3c6e99b67d514bd108c6c99

Package: firmware_dsp_sp2700, version: 2012-07-10_16.22_mcpre, status: n/a
File: bootflash:isr4400-universalk9.dir1/isr4400-firmware_dsp_sp2700-BLD-BLD_MCP_DEV_LATEST_
20120710_164422SSA.pkg, on: RP1/0
Built: 2012-07-10_16.22, by: mcpre
File SHA1 checksum: 8334565edf7843fe246783b1d5c6ed933d96d79e

Package: firmware_fpge, version: 2012-07-10_16.22_mcpre, status: n/a
```

Upgrading the Firmware on xDSL NIMs

To upgrade the firmware on a xDSL Network Interface Module (NIM), perform these steps:

Before you begin

When you boot the router in packages.conf mode with the Cisco IOS XE image (super package) during the installation period, you can upgrade or downgrade the firmware without reloading the router. You need to follow the steps described in Installing a Firmware Subpackage section before proceeding with the firmware upgrade.

If you do not boot the router in packages.conf mode with the Cisco IOS XE image, you need to follow the below prerequisites before proceeding with the firmware upgrade:

- Copy the firmware subpackage (NIM firmware) into bootflash:/mydir.

- Send a request to the platform software package expand file *boot flash:/mydir/<IOS-XE image>* to expand the super package.
- Reload the hardware module subslot to boot the module with the new firmware.
- Verify that the module is booted up with the new firmware using the **show platform software subslot x/y module firmware** command.

SUMMARY STEPS

1. copy Cisco IOS XE image into bootflash: **mydir**.
2. **request platform software package expand file** *bootflash:/mydir /<IOS-XE image* to expand super package.
3. **reload**.
4. **boot bootflash:mydir/ /packages.conf**.
5. **copy** NIM firmware subpackage to the folder **bootflash:mydir/**.
6. **request platform software package install** *rp 0 file bootflash:/mydir/<firmware subpackage>*.
7. **hw-module subslot x/y reload** to boot the module with the new firmware.
8. **show platform software subslot 0/2 module firmware** to verify that the module is booted up with the new firmware.

DETAILED STEPS

	Command or Action	Purpose
Step 1	copy Cisco IOS XE image into bootflash: **mydir**. **Example:** `Router# mkdir bootflash:mydir`	Creates a directory to save the expanded software image. You can use the same name as the image to name the directory.
Step 2	**request platform software package expand file** *bootflash:/mydir /<IOS-XE image* to expand super package. **Example:** `Router# request platform software package expand file bootflash:/mydir/isr4400-universalk9.03.14.00.S.155-1.S-std.SPA.bin`	Expands the platform software package to super package.

Step 3	**reload**.
	Example:
	`Router# reload`
	`rommon >`
		Enables ROMMON mode, which allows the software in the super package file to be activated.

Step 4	**boot bootflash:mydir/ /packages.conf**. **Example:** `rommon 1 > boot bootflash:mydir/packages.conf`	Boots the super package by specifying the path and name of the provisioning file: packages.conf.
Step 5	**copy** NIM firmware subpackage to the folder **bootflash:mydir/**. **Example:**	Copies the NIM firmware subpackage into bootflash:mydir.

	Command or Action	Purpose
	Router#**copy bootflash:isr4400-firmware_nim_xdsl.2014-11-17_11.05_39n.SSA.pkg bootflash:mydir/**	
Step 6	**request platform software package install** *rp 0 file bootflash:/mydir/<firmware subpackage>*. **Example:** Router#**equest platform software package install rp 0 file bootflash:mydir/isr4400-firmware_nim_xdsl.2014-11-17_11.05_39n.SSA.pkg**	Installs the software package.
Step 7	**hw-module subslot x/y reload** to boot the module with the new firmware. **Example:** Router#**hw-module subslot 0/2 reload**	Reloads the hardware module subslot and boots the module with the new firmware.
Step 8	**show platform software subslot 0/2 module firmware** to verify that the module is booted up with the new firmware. **Example:** Router# **show platform software subslot 0/2 module firmware** Pe	Displays the version of the newly installed firmware.

Examples

The following example shows how to perform firmware upgrade in a router module:

```
Routermkdir bootflash:mydir
Create directory filename [mydir]?
Created dir bootflash:/mydir
Router#c
Router#copy bootflash:isr4400-universalk9.03.14.00.S.155-1.S-std.SPA.bin bootflash:mydir/
Destination filename [mydir/isr4400-universalk9.03.14.00.S.155-1.S-std.SPA.bin]?
Copy in progress...CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCC
425288648 bytes copied in 44.826 secs (9487544 bytes/sec)
Router#
Router#
Router#dir bootflash:mydir
Directory of bootflash:/mydir/

632738  -rw-        425288648  Dec 12 2014 09:16:42 +00:00
isr4400-universalk9.03.14.00.S.155-1.S-std.SPA.bin

7451738112 bytes total (474025984 bytes free)
Router#

Router#request platform software package
expand file bootflash:/mydir/isr4400-universalk9.03.14.00.S.155-1.S-std.SPA.bin
```

```
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.

Router#reload
Proceed with reload? [confirm]

*Dec 12 09:26:09.874: %SYS-5-RELOAD: Reload requested by console. Reload Reason:
Reload Command.Dec 12 09:26:25.156 R0/0: %PMAN-5-EXITACTION: Process manager is exiting:
process exit with reload chassis code


Initializing Hardware ...

System integrity status: 00000610
 Rom image verified correctly
 System Bootstrap, Version 15.3(3r)S1, RELEASE SOFTWARE
Copyright (c) 1994-2013  by cisco Systems, Inc.

 Current image running: Boot ROM0

Last reset cause: LocalSoft
 Cisco ISR4451-X/K9 platform with 4194304 Kbytes of main memory


rommon 1  boot bootflash:mydir/packages.conf

 File size is 0x000028f1
 Located mydir/packages.conf
 Image size
10481 inode num 632741, bks cnt 3 blk size 8*512

#
 File size is 0x150ae3cc
 Located mydir/isr4400-mono-universalk9.03.14.00.S.155-1.S-std.SPA.pkg
 Image size 353035212 inode num 356929, bks cnt 86191 blk size 8*512
 ####################################################################
######################################################################
 Boot image size = 353035212 (0x150ae3cc) bytes

 Package header rev 1 structure detected
 Calculating SHA-1 hash...done
 validate_package: SHA-1 hash:
  calculated 8e966678:8afb08f4:8a88bb8f:fe591121:8bddf4b3
  expected   8e966678:8afb08f4:8a88bb8f:fe591121:8bddf4b3

 RSA Signed RELEASE Image Signature Verification Successful.
 Package Load Test Latency : 3799 msec
 Image validated
 Dec 12 09:28:50.338 R0/0: %FLASH_CHECK-3-DISK_QUOTA: Flash disk quota exceeded
[free space is 61864 kB] - Please clean up files on bootflash.

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

           cisco Systems, Inc.
           170 West Tasman Drive
```

```
            San Jose, California 95134-1706


Cisco IOS Software, ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Version 15.5(1)S,
RELEASE SOFTWARE (fc5)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2014 by Cisco Systems, Inc.
Compiled Thu 20-Nov-14 18:28 by mcpre



Cisco IOS-XE software, Copyright (c) 2005-2014 by cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

cisco ISR4451-X/K9 (2RU) processor with 1681388K/6147K bytes of memory.
Processor board ID FTX1736AJUT
2 Ethernet interfaces
4 Gigabit Ethernet interfaces
2 ATM interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of flash memory at bootflash:.

Press RETURN to get started!

*Dec 12 09:28:58.922:
%IOS_LICENSE_IMAGE_APPLICATION-6-LICENSE_LEVEL:
Module name = esg Next reboot level = appxk9 and License = appxk9
*Dec 12 09:28:58.943:
%IOS_LICENSE_IMAGE_APPLICATION-6-LICENSE_LEVEL:
Module name = esg Next reboot level = ipbasek9 and License = ipbasek9
*Dec 12 09:28:58.981:
 %ISR_THROUGHPUT-6-LEVEL: Throughput level has been set to 1000000 kbps
*Dec 12 09:29:13.302: %SPANTREE-5-EXTENDED_SYSID: Extended SysId enabled for type vlan
*Dec 12 09:29:14.142: %LINK-3-UPDOWN: Interface Lsmpi0, changed state to up
*Dec 12 09:29:14.142: %LINK-3-UPDOWN: Interface EOBC0, changed state to up
*Dec 12 09:29:14.142: %LINK-3-UPDOWN: Interface GigabitEthernet0, changed state to down
*Dec 12 09:29:14.142: %LINK-3-UPDOWN: Interface LIIN0, changed state to up
*Dec 12 09:28:51.438: %CMRP-3-PFU_MISSING:cmand:  The platform does not detect a power
```

```
supply in slot 1
*Dec 12 09:29:01.256: %CMLIB-6-THROUGHPUT_VALUE:cmand:  Throughput license found, throughput
 set to 1000000 kbps
*Dec 12 09:29:03.223: %CPPHA-7-START:cpp_ha:  CPP 0 preparing ucode
*Dec 12 09:29:03.238: %CPPHA-7-START:cpp_ha:  CPP 0 startup init
*Dec 12 09:29:11.335: %CPPHA-7-START:cpp_ha:  CPP 0 running init
*Dec 12 09:29:11.645: %CPPHA-7-READY:cpp_ha:  CPP 0 loading and initialization complete
*Dec 12 09:29:11.711: %IOSXE-6-PLATFORM:cpp_cp:
Process CPP_PFILTER_EA_EVENT__API_CALL__REGISTER
*Dec 12 09:29:16.280:
%IOSXE_MGMTVRF-6-CREATE_SUCCESS_INFO:
Management vrf Mgmt-intf created with ID 1, ipv4 table-id 0x1, ipv6 table-id 0x1E000001
*Dec 12 09:29:16.330:
%LINEPROTO-5-UPDOWN: Line protocol on Interface Lsmpi0, changed state to up
*Dec 12 09:29:16.330:
%LINEPROTO-5-UPDOWN: Line protocol on Interface EOBC0, changed state to up
*Dec 12 09:29:16.330:
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0, changed state to down
*Dec 12 09:29:16.330:
%LINEPROTO-5-UPDOWN: Line protocol on Interface LIIN0, changed state to up
*Dec 12 09:29:17.521: %SYS-5-LOG_CONFIG_CHANGE: Buffer logging disabled
*Dec 12 09:29:18.867: %SYS-5-CONFIG_I: Configured from memory by console
*Dec 12 09:29:18.870:
%IOSXE_OIR-6-REMSPA: SPA removed from subslot 0/0, interfaces disabled
*Dec 12 09:29:18.870:
%IOSXE_OIR-6-REMSPA: SPA removed from subslot 0/1, interfaces disabled
*Dec 12 09:29:18.871:
%IOSXE_OIR-6-REMSPA: SPA removed from subslot 0/2, interfaces disabled
*Dec 12 09:29:18.873:
%SPA_OIR-6-OFFLINECARD: SPA (ISR4451-X-4x1GE) offline in subslot 0/0
*Dec 12 09:29:18.874: %SPA_OIR-6-OFFLINECARD: SPA (NIM-VA-B) offline in subslot 0/1
*Dec 12 09:29:18.874: %SPA_OIR-6-OFFLINECARD: SPA (NIM-VAB-A) offline in subslot 0/2
*Dec 12 09:29:18.876: %IOSXE_OIR-6-INSCARD: Card (fp) inserted in slot F0
*Dec 12 09:29:18.876: %IOSXE_OIR-6-ONLINECARD: Card (fp) online in slot F0
*Dec 12 09:29:18.882: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 0/0
*Dec 12 09:29:18.884: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 0/1
*Dec 12 09:29:18.884: %IOSXE_OIR-6-INSSPA: SPA inserted in subslot 0/2
*Dec 12 09:29:18.935: %SYS-5-RESTART: System restarted --
Cisco IOS Software, ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Version 15.5(1)S,
RELEASE SOFTWARE (fc5)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2014 by Cisco Systems, Inc.
Compiled Thu 20-Nov-14 18:28 by mcpre
*Dec 12 09:29:18.895: %SPA-3-ENVMON_NOT_MONITORED:iomd:  Environmental monitoring
is not enabled for ISR4451-X-4x1GE[0/0]
*Dec 12 09:29:19.878: %LINK-5-CHANGED: Interface GigabitEthernet0,

changed state to administratively down
*Dec 12 09:29:22.419: %SPA_OIR-6-ONLINECARD: SPA (ISR4451-X-4x1GE) online in subslot 0/0
*Dec 12 09:29:22.610: %SYS-6-BOOTTIME: Time taken to reboot after reload =  194 seconds
*Dec 12 09:29:24.354: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/0,
changed state to down
*Dec 12 09:29:24.415: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/2,
changed state to down
*Dec 12 09:29:24.417: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/3,
changed state to down
*Dec 12 09:29:30.919: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/0,
changed state to up
*Dec 12 09:29:30.925: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/2,
 changed state to up
*Dec 12 09:29:30.936: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/3,
 changed state to up
*Dec 12 09:29:31.919: %LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/0, changed state to up
```

```
*Dec 12 09:29:31.930: %LINEPROTO-5-UPDOWN: Line protocol on
 Interface GigabitEthernet0/0/2, changed state to up
*Dec 12 09:29:31.936: %LINEPROTO-5-UPDOWN: Line protocol on
Interface GigabitEthernet0/0/3, changed state to up
*Dec 12 09:29:34.147: %SSH-5-ENABLED: SSH 1.99 has been enabled
*Dec 12 09:30:29.152: %SPA_OIR-6-ONLINECARD: SPA (NIM-VA-B) online in subslot 0/1
*Dec 12 09:30:29.470: %SPA_OIR-6-ONLINECARD: SPA (NIM-VAB-A) online in subslot 0/2
*Dec 12 09:30:31.152: %LINK-3-UPDOWN: Interface Ethernet0/1/0, changed state to down
*Dec 12 09:30:31.152: %LINK-3-UPDOWN: Interface ATM0/1/0, changed state to down
*Dec 12 09:30:31.470: %LINK-3-UPDOWN: Interface Ethernet0/2/0, changed state to down
*Dec 12 09:30:31.470: %LINK-3-UPDOWN: Interface ATM0/2/0, changed state to down
*Dec 12 09:31:03.074: %CONTROLLER-5-UPDOWN: Controller VDSL 0/2/0, changed state to up
*Dec 12 09:31:05.075: %LINK-3-UPDOWN: Interface Ethernet0/2/0, changed state to up
*Dec 12 09:31:06.076: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/2/0,
changed state to up
*Dec 12 09:31:12.559: %CONTROLLER-5-UPDOWN: Controller VDSL 0/1/0, changed state to up
*Dec 12 09:31:20.188: %LINK-3-UPDOWN: Interface ATM0/1/0, changed state to up
*Dec 12 09:31:21.188: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM0/1/0,
changed state to up
Router>
Router>en
Password:
Router#
Router#show controller vdsl 0/2/0
Controller VDSL 0/2/0 is UP

Daemon Status:  UP

   XTU-R (DS)  XTU-C (US)
Chip Vendor ID:  'BDCM'    'BDCM'
Chip Vendor Specific:   0x0000    0xA41B
Chip Vendor Country:    0xB500    0xB500
Modem Vendor ID: 'CSCO'    '    '
Modem Vendor Specific:  0x4602    0x0000
Modem Vendor Country:   0xB500    0x0000
Serial Number Near:     FOC18426DQ8 4451-X/K15.5(1)S
Serial Number Far:
Modem Version Near:     15.5(1)S
Modem Version Far:      0xa41b

Modem Status(L1): TC Sync (Showtime!)
DSL Config Mode: VDSL2
Trained Mode(L1): G.993.2 (VDSL2) Profile 30a

TC Mode:  PTM
Selftest Result: 0x00
DELT configuration: disabled
DELT state:  not running

Failed full inits: 0
Short inits:  0
Failed short inits: 0

Modem FW  Version: 4.14L.04
Modem PHY Version: A2pv6F039h.d24o_rc1

Line 1:

   XTU-R (DS)  XTU-C (US)
Trellis:   ON    ON
SRA:     disabled   disabled
SRA count:    0    0
Bit swap:    enabled   enabled
Bit swap count:   9    0
```

```
Profile 30a:    enabled
Line Attenuation:   3.5 dB    0.0 dB
Signal Attenuation:   0.0 dB    0.0 dB
Noise Margin:   30.9 dB  12.4 dB
Attainable Rate: 200000 kbits/s   121186 kbits/s
Actual Power:  13.3 dBm   7.2 dBm
Per Band Status:        D1  D2  D3  U0  U1  U2  U3
Line Attenuation(dB):   0.9 1.5 5.5 N/A 0.1 0.9 3.8
Signal Attenuation(dB): 0.8 1.5 5.5 N/A 0.0 0.2 3.2
Noise Margin(dB):       31.1 31.0 30.9 N/A 12.3 12.4 12.5
Total FECC:  0    0
Total ES:  0    0
Total SES:  0    0
Total LOSS:  0    0
Total UAS:  51    51
Total LPRS:  0    0
Total LOFS:  0    0
Total LOLS:  0    0


     DS Channel1   DS Channel0 US Channel1   US Channel0
Speed (kbps):    NA       100014  NA       100014
SRA Previous Speed:   NA            0  NA            0
Previous Speed:    NA            0  NA            0
Reed-Solomon EC:   NA            0  NA            0
CRC Errors:    NA            0  NA            0
Header Errors:    NA            0  NA            0
Interleave (ms):   NA         9.00  NA         0.00
Actual INP:    NA         4.00  NA         0.00

Training Log : Stopped
Training Log Filename : flash:vdsllog.bin

Router#
Router#

Router#copy bootflash:isr4400-firmware_nim_xdsl.2014-11-17_11.05_39n.SSA.pkg
 bootflash:mydir/
Destination filename [mydir/isr4400-firmware_nim_xdsl.2014-11-17_11.05_39n.SSA.pkg]?
Copy in progress...CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
6640604 bytes copied in 1.365 secs (4864911 bytes/sec)
Router#

Router#request platform software package install rp 0 file
bootflash:mydir/isr4400-firmware_nim_xdsl.2014-11-17_11.05_39n.SSA.pkg
--- Starting local lock acquisition on R0 ---
Finished local lock acquisition on R0

--- Starting file path checking ---
Finished file path checking

--- Starting image file verification ---
Checking image file names
Locating image files and validating name syntax
  Found isr4400-firmware_nim_xdsl.2014-11-17_11.05_39n.SSA.pkg
Verifying image file locations
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification

--- Starting candidate package set construction ---
Verifying existing software set
```

```
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Checking if resulting candidate package set would be complete
Finished candidate package set construction

--- Starting ISSU compatiblity verficiation ---
Verifying image type compatibility
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking package specific compatibility
Finished ISSU compatiblity verficiation

--- Starting impact testing ---
Checking operational impact of change
Finished impact testing

--- Starting list of software package changes ---
Old files list:
  Removed isr4400-firmware_nim_xdsl.03.14.00.S.155-1.S-std.SPA.pkg
New files list:
  Added isr4400-firmware_nim_xdsl.2014-11-17_11.05_39n.SSA.pkg
Finished list of software package changes

--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes

--- Starting analysis of software changes ---
Finished analysis of software changes

--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Finding latest command shortlist lookup file
  Finding latest command shortlist file
  Assembling CLI output libraries
  Assembling CLI input libraries
Skipping soft links for firmware upgrade
Skipping soft links for firmware upgrade
  Assembling Dynamic configuration files
  Applying interim IPC and database definitions
rsync: getaddrinfo: cc2-0 873: Name or service not known rsync error:
error in socket IO (code 10) at /auto/mcpbuilds19/
release/03.14.00.S/BLD-V03_14_00_S_FC5/contrib/rsync/clientserver.c(104) [sender=2.6.9]
rsync: getaddrinfo: cc2-0 873: Name or service not known rsync error:
error in socket IO (code 10) at /auto/mcpbuilds19/
release/03.14.00.S/BLD-V03_14_00_S_FC5/contrib/rsync/clientserver.c(104) [sender=2.6.9]
rsync: getaddrinfo: cc2-0 873: Name or service not known rsync error:
error in socket IO (code 10) at /auto/mcpbuilds19
/release/03.14.00.S/BLD-V03_14_00_S_FC5/contrib/rsync/clientserver.c(104) [sender=2.6.9]
  Replacing running software
  Replacing CLI software
  Restarting software
  Applying final IPC and database definitions
rsync: getaddrinfo: cc2-0 873: Name or service not known rsync error:
error in socket IO (code 10) at /auto/mcpbuilds19/
release/03.14.00.S/BLD-V03_14_00_S_FC5/contrib/rsync/clientserver.c(104) [sender=2.6.9]
```

```
  Generating software version information
  Notifying running software of updates
  Unblocking peer synchronization of operating information
Unmounting old packages
Cleaning temporary installation files
  Finished update running software

SUCCESS: Finished installing software.
Router#
Router#show platform software subslot 0/2 module firmware
Avg Load info
-------------------------------------------
1.83 1.78 1.44 3/45 607

Kernel distribution info
-------------------------------------------
Linux version 3.4.11-rt19 (sapanwar@blr-atg-001) (gcc version 4.6.2
(Buildroot 2011.11) ) #3 SMP PREEMPT Fri Nov 7 09:26:19 IST 2014

Module firmware versions
-------------------------------------------
Modem Fw Version: 4.14L.04
Modem Phy Version: A2pv6F039h.d24o_rc1

Boot Loader: Secondry
-------------------------------------------
Version: 1.1

Modem Up time
-------------------------------------------
0D 0H 25M 38S

Router#

Router#hw-module subslot 0/2 reload
Proceed with reload of module? [confirm]
Router#
*Dec 12 09:55:59.645: %IOSXE_OIR-6-SOFT_RELOADSPA: SPA(NIM-VAB-A)
reloaded on subslot 0/2
*Dec 12 09:55:59.646: %SPA_OIR-6-OFFLINECARD: SPA (NIM-VAB-A) offline in subslot 0/2
*Dec 12 09:55:59.647: %CONTROLLER-5-UPDOWN: Controller VDSL 0/2/0, changed state to down
*Dec 12 09:57:22.514: new extended attributes received from iomd(slot 0 bay 2 board 0)
*Dec 12 09:57:22.514: %IOSXE_OIR-6-SOFT_RELOADSPA: SPA(NIM-VAB-A)
 reloaded on subslot 0/2
*Dec 12 09:57:22.515: %SPA_OIR-6-OFFLINECARD: SPA (NIM-VAB-A) offline in subslot 0/2
Router#
Router#
*Dec 12 09:58:35.471: %SPA_OIR-6-ONLINECARD: SPA (NIM-VAB-A) online in subslot 0/2
*Dec 12 09:58:37.470: %LINK-3-UPDOWN: Interface Ethernet0/2/0, changed state to down
*Dec 12 09:58:37.470: %LINK-3-UPDOWN: Interface ATM0/2/0, changed state to down
Router#

Router#show platform software subslot 0/2 module firmware
Avg Load info
-------------------------------------------
0.84 0.23 0.08 1/45 598

Kernel distribution info
-------------------------------------------
Linux version 3.4.11-rt19 (sapanwar@blr-atg-001) (gcc version 4.6.2 (Buildroot 2011.11) )
#6 SMP PREEMPT Mon Nov 17 10:51:41 IST 2014

Module firmware versions
-------------------------------------------
```

```
Modem Fw Version: 4.14L.04
Modem Phy Version: A2pv6F039n.d24o_rc1

Boot Loader: Secondry
-------------------------------------------
Version: 1.1

Modem Up time
-------------------------------------------
0D 0H 0M 42S

Router#
```

CHAPTER 9

Installing the Software using install Commands

Installing the Software Using install Commands

From Cisco IOS XE 17.15.1a, all Cisco IOS XE platforms are shipped in install mode by default. Users can boot the platform, and upgrade or downgrade to Cisco IOS XE software versions using a set of **install** commands.

Restrictions for Installing the Software Using install Commands

- ISSU is not covered in this feature.
- Install mode requires a reboot of the system.

Information About Installing the Software Using install Commands

For routers shipped in install mode, a set of **install** commands can be used for starting, upgrading and downgrading of platforms in install mode. This update is applicable to the Cisco Catalyst 8000 Edge platforms.

From Cisco IOS XE 17.15.1a release, this update is applicable to all Cisco IOS XE platforms.

The following table describes the differences between Bundle mode and Install mode:

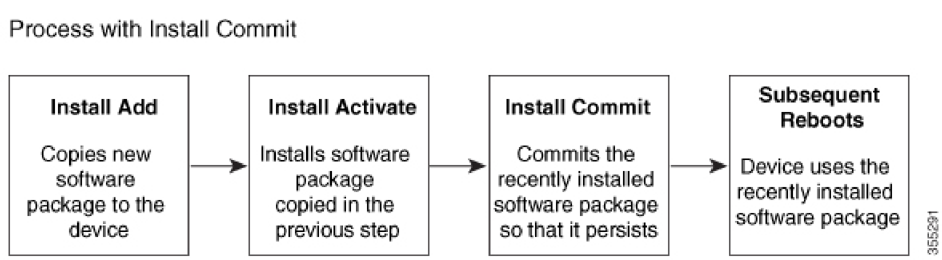
Table 17: Bundle Mode vs Install Mode

Bundle Mode	Install Mode
This mode provides a consolidated boot process, using local (hard disk, flash) or remote (TFTP) .bin image. **Note** Bundle boot from USB and TFTPBoot is not supported.	This mode uses the local (bootflash) packages.conf file for the boot process.
This mode uses a single .bin file.	.bin file is replaced with expanded .pkg files in this mode.
CLI: `#boot system file <filename>`	CLI: `#install add file bootflash: [activate commit]`
To upgrade in this mode, point the boot system to the new image.	To upgrade in this mode, use the **install** commands.
Image Auto-Upgrade: When a new Field-Replaceable Unit (FRU) is inserted in a modular chassis, manual intervention is required to get the new FRU running with the same version as the active FRUs.	Image Auto-Upgrade: When a new FRU is inserted in a modular chassis, the joining FRU is auto-upgraded to the image version in sync with the active FRUs.
Rollback: Rollback to the previous image with multiple Software Maintenance Updates (SMUs) may require multiple reloads.	Rollback: Enables rollback to an earlier version of Cisco IOS XE software, including multiple patches in single reload.

Install Mode Process Flow

The install mode process flow comprises three commands to perform installation and upgrade of software on platforms–**install add**, **install activate**, and **install commit**.

The following flow chart explains the install process with **install** commands:



The **install add** command copies the software package from a local or remote location to the platform. The location can be FTP, HTTP, HTTPs, or TFTP. The command extracts individual components of the .package file into subpackages and packages.conf files. It also validates the file to ensure that the image file is specific to the platform on which it is being installed.

The **install activate** command performs the required validations and provisions the packages previously added using the **install add** command. It also triggers a system reload.

The **install commit** command confirms the packages previously activated using the **install activate** command, and makes the updates persistent over reloads.

> **Note** Installing an update replaces any previously installed software image. At any time, only one image can be installed in a device.

The following set of install commands is available:

Table 18: List of install Commands

Command	Syntax	Purpose
install add	**install add file** *location:filename.bin*	Copies the contents of the image, package, and SMUs to the software repository. File location may be local or remote. This command does the following: • Validates the file–checksum, platform compatibility checks, and so on. • Extracts individual components of the package into subpackages and packages.conf • Copies the image into the local inventory and makes it available for the next steps.
install activate	**install activate**	Activates the package added using the **install add** command. • Use the **show install summary** command to see which image is inactive. This image will get activated. • System reloads on executing this command. Confirm if you want to proceed with the activation. Use this command with the **prompt-level none** keyword to automatically ignore any confirmation prompts.

Command	Syntax	Purpose
(install activate) auto abort-timer	**install activate auto-abort timer** *<30-1200>*	The **auto-abort timer** starts automatically, with a default value of 120 minutes. If the **install commit** command is not executed within the time provided, the activation process is terminated, and the system returns to the last-committed state. • You can change the time value while executing the **install activate** command. • The **install commit** command stops the timer, and continues the installation process. • The **install activate auto-abort timer stop** command stops the timer without committing the package. • Use this command with the **prompt-level none** keyword to automatically ignore any confirmation prompts. • This command is valid only in the three-step install variant.
install commit	**install commit**	Commits the package activated using the **install activate** command, and makes it persistent over reloads. • Use the **show install summary** command to see which image is uncommitted. This image will get committed.

Command	Syntax	Purpose
install abort	**install abort**	Terminates the installation and returns the system to the last-committed state. • This command is applicable only when the package is in activated status (uncommitted state). • If you have already committed the image using the **install commit** command, use the **install rollback to** command to return to the preferred version.
install remove	**install remove {file** *<filename>* **| inactive}**	Deletes inactive packages from the platform repository. Use this command to free up space. • **file**: Removes specified files. • **inactive**: Removes all the inactive files.
install rollback to	**install rollback to {base | label | committed | id}**	Rolls back the software set to a saved installation point or to the last-committed installation point. The following are the characteristics of this command: • Requires reload. • Is applicable only when the package is in committed state. • Use this command with the **prompt-level none** keyword to automatically ignore any confirmation prompts. **Note** If you are performing install rollback to a previous image, the previous image must be installed in install mode. Only SMU rollback is possible in bundle mode.

Command	Syntax	Purpose
install deactivate	**install deactivate file** <*filename*>	Removes a package from the platform repository. This command is supported only for SMUs. • Use this command with the **prompt-level none** keyword to automatically ignore any confirmation prompts.

The following show commands are also available:

Table 19: List of show Commands

Command	Syntax	Purpose
show install log	**show install log**	Provides the history and details of all install operations that have been performed since the platform was booted.
show install package	**show install package** <*filename*>	Provides details about the .pkg/.bin file that is specified.
show install summary	**show install summary**	Provides an overview of the image versions and their corresponding install states for all the FRUs. • The table that is displayed will state for which FRUs this information is applicable. • If all the FRUs are in sync in terms of the images present and their state, only one table is displayed. • If, however, there is a difference in the image or state information among the FRUs, each FRU that differs from the rest of the stack is listed in a separate table.
show install active	**show install active**	Provides information about the active packages for all the FRUs. If there is a difference in the information among the FRUs, each FRU that differs from the rest of the stack is listed in a separate table.

Command	Syntax	Purpose
show install inactive	**show install inactive**	Provides information about the inactive packages, if any, for all the FRUs. If there is a difference in the information among the FRUs, each FRU that differs from the rest of the stack is listed in a separate table.
show install committed	**show install committed**	Provides information about the committed packages for all the FRUs. If there is a difference in the information among the FRUs, each FRU that differs from the rest of the stack is listed in a separate table.
show install uncommitted	**show install uncommitted**	Provides information about uncommitted packages, if any, for all the FRUs. If there is a difference in the information among the FRUs, each FRU that differs from the rest of the stack is listed in a separate table.
show install rollback	**show install rollback {point-id | label}**	Displays the package associated with a saved installation point.
show version	**show version [rp-slot] [installed [user-interface] | provisioned | running]**	Displays information about the current package, along with hardware and platform information.

Booting the Platform in Install Mode

You can install, activate, and commit a software package using a single command (one-step install) or multiple separate commands (three-step install).

If the platform is working in bundle mode, the one-step install procedure must be used to initially convert the platform from bundle mode to install mode. Subsequent installs and upgrades on the platform can be done with either one-step or three-step variants.

One-Step Installation or Converting from Bundle Mode to Install Mode

> **Note**
> - All the CLI actions (for example, add, activate, and so on) are executed on all the available FRUs.
> - The configuration save prompt will appear if an unsaved configuration is detected.
> - The reload prompt will appear after the second step in this workflow. Use the **prompt-level none** keyword to automatically ignore the confirmation prompts.
> - If the prompt-level is set to None, and there is an unsaved configuration, the install fails. You must save the configuration before reissuing the command.

Use the one-step install procedure described below to convert a platform running in bundle boot mode to install mode. After the command is executed, the platform reboots in install boot mode.

Later, the one-step install procedure can also be used to upgrade the platform.

This procedure uses the **install add file activate commit** command in privileged EXEC mode to install a software package, and to upgrade the platform to a new version.

SUMMARY STEPS

1. **enable**
2. **install add file location:** *filename* [**activate commit**]
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device>enable`	Enables privileged EXEC mode. Enter your password, if prompted.

Step 2	**install add file location:** *filename* [**activate commit**]
	Example:
	`Device#install add file`
	`bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.bin`
	` activate commit`
		Copies the software install package from a local or remote location (through FTP, HTTP, HTTPs, or TFTP) to the platform and extracts the individual components of the .package file into subpackages and packages.conf files. It also performs a validation and compatibility check for the platform and image versions, activates the package, and commits the package to make it persistent across reloads.
		The platform reloads after this command is run.

Step 3	**exit** **Example:** `Device#exit`	Exits privileged EXEC mode and returns to user EXEC mode.

Three-Step Installation

> **Note**
> - All the CLI actions (for example, add, activate, and so on) are executed on all the available FRUs.
> - The configuration save prompt will appear if an unsaved configuration is detected.
> - The reload prompt will appear after the install activate step in this workflow. Use the **prompt-level none** keyword to automatically ignore the confirmation prompts.

The three-step installation procedure can be used only after the platform is in install mode. This option provides more flexibility and control to the customer during installation.

This procedure uses individual **install add**, **install activate**, and **install commit** commands for installing a software package, and to upgrade the platform to a new version.

SUMMARY STEPS

1. **enable**
2. **install add file location:** *filename*
3. **show install summary**
4. **install activate** [**auto-abort-timer** <*time*>]
5. **install abort**
6. **install commit**
7. **install rollback to committed**
8. **install remove** {**file** *filesystem: filename* | **inactive**}
9. **show install summary**
10. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device>enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**install add file location:** *filename* **Example:** `Device#install add file bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.bin`	Copies the software install package from a remote location (through FTP, HTTP, HTTPs, or TFTP) to the platform, and extracts the individual components of the .package file into subpackages and packages.conf files.
Step 3	**show install summary** **Example:** `Device#show install summary`	(Optional) Provides an overview of the image versions and their corresponding install state for all the FRUs.

	Command or Action	Purpose
Step 4	**install activate** [**auto-abort-timer** <*time*>] **Example:** `Device# install activate auto-abort-timer 120`	Activates the previously added package and reloads the platform. • When doing a full software install, do not provide a package filename. • In the three-step variant, **auto-abort-timer** starts automatically with the **install activate** command; the default for the timer is 120 minutes. If the **install commit** command is not run before the timer expires, the install process is automatically terminated. The platform reloads and boots up with the last committed version.
Step 5	**install abort** **Example:** `Device#install abort`	(Optional) Terminates the software install activation and returns the platform to the last committed version. • Use this command only when the image is in activated state, and not when the image is in committed state.
Step 6	**install commit** **Example:** `Device#install commit`	Commits the new package installation and makes the changes persistent over reloads.
Step 7	**install rollback to committed** **Example:** `Device#install rollback to committed`	(Optional) Rolls back the platform to the last committed state.
Step 8	**install remove** {**file** *filesystem: filename* | **inactive**} **Example:** `Device#install remove inactive`	(Optional) Deletes software installation files. • **file**: Deletes a specific file • **inactive**: Deletes all the unused and inactive installation files.
Step 9	**show install summary** **Example:** `Device#show install summary`	(Optional) Displays information about the current state of the system. The output of this command varies according to the **install** commands run prior to this command.
Step 10	**exit** **Example:** `Device#exit`	Exits privileged EXEC mode and returns to user EXEC mode.

Upgrading in Install Mode

Use either the one-step installation or the three-step installation to upgrade the platform in install mode.

Downgrading in Install Mode

Use the **install rollback** command to downgrade the platform to a previous version by pointing it to the appropriate image, provided the image you are downgrading to was installed in install mode.

The **install rollback** command reloads the platform and boots it with the previous image.

Note The **install rollback** command succeeds only if you have not removed the previous file using the **install remove inactive** command.

Alternatively, you can downgrade by installing the older image using the **install** commands.

Terminating a Software Installation

You can terminate the activation of a software package in the following ways:

- When the platform reloads after activating a new image, the auto-abort-timer is triggered (in the three-step install variant). If the timer expires before issuing the **install commit** command, the installation process is terminated, and the platform reloads and boots with the last committed version of the software image.

  Alternatively, use the **install auto-abort-timer stop** command to stop this timer, without using the **install commit** command. The new image remains uncommitted in this process.

- Using the **install abort** command returns the platform to the version that was running before installing the new software. Use this command before issuing the **install commit** command.

Configuration Examples for Installing the Software Using install Commands

The following is an example of the one-step installation or converting from bundle mode to install mode:

```
Router# install add file
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.bin
 activate commit
install_add_activate_commit: START Thu Oct 28 21:57:21 UTC 2021

System configuration has been modified.
Press Yes(y) to save the configuration and proceed.
Press No(n) for proceeding without saving the configuration.
Press Quit(q) to exit, you may save configuration and re-enter the command. [y/n/q]y
Building configuration...

[OK]Modified configuration has been saved

*Oct 28 21:57:39.818: %SYS-6-PRIVCFG_ENCRYPT_SUCCESS: Successfully encrypted private config
 file
*Oct 28 21:57:39.925: %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started install
 one-shot
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.bininstall_add_activate_commit:
 Adding PACKAGE
```

```
install_add_activate_commit: Checking whether new add is allowed ....

--- Starting Add ---
Performing Add on Active/Standby
  [1] Add package(s) on R0
  [1] Finished Add on R0
Checking status of Add on [R0]
Add: Passed on [R0]
Finished Add

Image added. Version: 17.07.01.0.1515
install_add_activate_commit: Activating PACKAGE
Following packages shall be activated:
/bootflash/c8000be-rpboot.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-mono-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_sm_nim_adpt.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_sm_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_sm_async.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_sm_1t3e3.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_sm_10g.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_prince.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_nim_xdsl.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_nim_ssd.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_nim_shdsl.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_nim_ge.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_nim_cwan.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_nim_bri_st_fw.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_nim_async.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_ngwic_t1e1.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_dsp_tilegx.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_dsp_analogbri.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
/bootflash/c8000be-firmware_dreamliner.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg

This operation may require a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on Active/Standby

*Oct 28 22:05:49.484: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: R0/0: rollback_timer:
Install auto abort timer will expire in 7200 seconds  [1] Activate package(s) on R0
  [1] Finished Activate on R0
Checking status of Activate on [R0]
Activate: Passed on [R0]
Finished Activate

--- Starting Commit ---
Performing Commit on Active/Standby
  [1] Commit package(s) on R0

Building configuration...
  [1] Finished Commit on R0
Checking status of Commit on [R0]
Commit: Passed on [R0]
Finished Commit

[OK]
*Oct 28 22:06:55.375: %SYS-6-PRIVCFG_ENCRYPT_SUCCESS: Successfully encrypted private config
 fileSend model notification for install_add_activate_commit before reload
Install will reload the system now!
SUCCESS: install_add_activate_commit  Thu Oct 28 22:07:22 UTC 2021

Router#
*Oct 28 22:07:22.661: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install one-shot PACKAGE
```

```
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.binOct
 28 22:07:26.864: %PMAN-5-EXITACTION: R0/0: pvp: Process manager is exiting: reload action
 requested

□

Press RETURN to get started!
```

The following is an example of the three-step installation:

```
Router# install add file
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.bin

install_add: START Thu Oct 28 22:36:43 UTC 2021

*Oct 28 22:36:44.526: %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started install
 add
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.bininstall_add:
 Adding PACKAGE
install_add: Checking whether new add is allowed ....

--- Starting Add ---
Performing Add on Active/Standby
  [1] Add package(s) on R0
  [1] Finished Add on R0
Checking status of Add on [R0]
Add: Passed on [R0]
Finished Add

Image added. Version: 17.07.01.0.1601
SUCCESS: install_add  Thu Oct 28 22:40:25 UTC 2021

Router#
*Oct 28 22:40:25.971: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install add PACKAGE
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.bin

Router# show install log
[0|install_op_boot]: START Thu Oct 28 22:09:29 Universal 2021
[0|install_op_boot(INFO, )]: Mount IMG INI state base image
[0|install_op_boot]: END SUCCESS  Thu Oct 28 22:09:30 Universal 2021
[0|install_op_boot(INFO, )]: cleanup_trap  remote_invocation 0 operation install_op_boot
.. 0 .. 0
[1|display_install_log]: START Thu Oct 28 22:12:11 UTC 2021
[2|install_add]: START Thu Oct 28 22:36:43 UTC 2021
[2|install_add(INFO, )]: Set INSTALL_TYPE to PACKAGE
[2|install_add(CONSOLE, )]: Adding PACKAGE
[2|install_add(CONSOLE, )]: Checking whether new add is allowed ....
[2|install_add(INFO, )]: check_add_op_allowed: Install type PACKAGE
[remote|install_add]: START Thu Oct 28 22:37:12 UTC 2021
[remote|install_add]: END SUCCESS  Thu Oct 28 22:40:10 UTC 2021
[remote|install_add(INFO, )]: cleanup_trap  remote_invocation 1 operation install_add .. 0
 .. 0
[2|install_add(INFO, )]: Remote output from R0
[2|install_add(INFO, )]: install_add: START Thu Oct 28 22:37:12 UTC 2021
Expanding image file:
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.bin
Verifying parameters
Expanding superpackage
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.bin
... parameters verified
Validating package type
... package type validated
```

```
Copying package files
    c8000be-firmware_dreamliner.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

c8000be-firmware_dsp_analogbri.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

    c8000be-firmware_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

    c8000be-firmware_dsp_tilegx.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

    c8000be-firmware_ngwic_t1e1.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

    c8000be-firmware_nim_async.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

c8000be-firmware_nim_bri_st_fw.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

     c8000be-firmware_nim_cwan.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
     c8000be-firmware_nim_ge.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
    c8000be-firmware_nim_shdsl.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

     c8000be-firmware_nim_ssd.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
     c8000be-firmware_nim_xdsl.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
     c8000be-firmware_prince.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
     c8000be-firmware_sm_10g.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
     c8000be-firmware_sm_1t3e3.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
     c8000be-firmware_sm_async.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

c8000be-firmware_sm_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

    c8000be-firmware_sm_nim_adpt.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

     c8000be-mono-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
     c8000be-rpboot.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
  WARNING: A different version of provisioning file packages.conf already exists in bootflash:

  WARNING: The provisioning file from the expanded bundle will be saved as
  WARNING: bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211027_0.conf
... package files copied
SUCCESS: Finished expanding all-in-one software package.
Image file expanded
SUCCESS: install_add  Thu Oct 28 22:40:10 UTC 2021
[2|install_add]: END SUCCESS  Thu Oct 28 22:40:25 UTC 2021
[2|install_add(INFO, )]: cleanup_trap  remote_invocation 0 operation install_add .. 0 .. 0
[3|COMP_CHECK]: START Thu Oct 28 22:40:26 UTC 2021
[3|COMP_CHECK]: END FAILED exit(1)  Thu Oct 28 22:40:27 UTC 2021
[3|COMP_CHECK(INFO, )]: cleanup_trap  remote_invocation 0 operation COMP_CHECK .. 1 .. 1
[4|install_activate]: START Thu Oct 28 22:42:53 UTC 2021
[4|install_activate(INFO, require user prompt)]: install_cli
[4|install_activate(CONSOLE, )]: Activating PACKAGE
[4|install_activate(INFO, )]: Acquiring transaction lock...
[4|install_activate(INFO, )]: global_trans_lock:
/bootflash/.installer/install_global_trans_lock
[4|install_activate(INFO, )]: tmp_global_trans_lock: /tmp/tmp_install_global_trans_lock
[4|install_activate(INFO, )]: tmp lock does not exist: /tmp/tmp_install_global_trans_lock
[4|install_activate(INFO, )]: global_trans_lock:
/bootflash/.installer/install_global_trans_lock
[4|install_activate(INFO, )]: tmp_global_trans_lock: /tmp/tmp_install_global_trans_lock
[4|install_activate(INFO, )]: local_trans_lock: /bootflash/.installer/install_local_trans_lock
[4|install_activate(INFO, )]: global_trans_lock:
/bootflash/.installer/install_global_trans_lock
[4|install_activate(INFO, )]: validate_lock: lock_duration is 7200
[4|install_activate(INFO, )]: install type stored in lock PACKAGE, install type PACKAGE,
install operation install_activate
```

```
[4|install_activate(INFO, )]: lock duration: 7200
[4|install_activate(INFO, )]: extend trans lock done.
/bootflash/.installer/install_global_trans_lock
[4|install_activate(INFO, require user prompt)]: install_cli
[4|install_activate( FATAL)]: Cannot proceed activate because of user input
[4|install_activate(INFO, )]: cleanup_trap  remote_invocation 0 operation install_activate
 .. 6 .. 0
[5|install_add]: START Thu Oct 28 22:45:48 UTC 2021
[5|install_add(INFO, )]: Set INSTALL_TYPE to PACKAGE
[5|install_add(CONSOLE, )]: Adding PACKAGE
[5|install_add(CONSOLE, )]: Checking whether new add is allowed ....
[5|install_add(INFO, )]: check_add_op_allowed: Install type PACKAGE
[5|install_add( FATAL)]: Super package already added. Add operation not allowed. install
remove inactive can be used to discard added packages

Router# install activate
install_activate: START Thu Oct 28 23:57:57 UTC 2021
install_activate: Activating PACKAGE

*Oct 28 23:57:57.823: %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started install
 activateFollowing packages shall be activated:
/bootflash/c8000be-rpboot.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-mono-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_sm_nim_adpt.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_sm_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_sm_async.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_sm_1t3e3.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_sm_10g.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_prince.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_nim_xdsl.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_nim_ssd.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_nim_shdsl.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_nim_ge.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_nim_cwan.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_nim_bri_st_fw.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_nim_async.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_ngwic_t1e1.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_dsp_tilegx.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_dsp_analogbri.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
/bootflash/c8000be-firmware_dreamliner.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

This operation may require a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on Active/Standby

*Oct 29 00:04:19.400: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: R0/0: rollback_timer:
Install auto abort timer will expire in 7200 seconds  [1] Activate package(s) on R0
    --- Starting list of software package changes ---
    Old files list:
      Modified
c8000be-firmware_dreamliner.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_dsp_analogbri.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg

      Modified
c8000be-firmware_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_dsp_tilegx.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_ngwic_t1e1.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_nim_async.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
```

```
c8000be-firmware_nim_bri_st_fw.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg

      Modified
c8000be-firmware_nim_cwan.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_nim_ge.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_nim_shdsl.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_nim_ssd.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_nim_xdsl.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_prince.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_sm_10g.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_sm_1t3e3.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_sm_async.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-firmware_sm_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg

      Modified
c8000be-firmware_sm_nim_adpt.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified
c8000be-mono-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
      Modified c8000be-rpboot.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
    New files list:
      Added
c8000be-firmware_dreamliner.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_dsp_analogbri.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

      Added
c8000be-firmware_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_dsp_tilegx.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_ngwic_t1e1.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_nim_async.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_nim_bri_st_fw.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

      Added
c8000be-firmware_nim_cwan.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_nim_ge.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_nim_shdsl.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_nim_ssd.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_nim_xdsl.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_prince.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_sm_10g.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_sm_1t3e3.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-firmware_sm_async.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
```

```
c8000be-firmware_sm_dsp_sp2700.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg

      Added
c8000be-firmware_sm_nim_adpt.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added
c8000be-mono-universalk9.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
      Added c8000be-rpboot.BLD_V177_THROTTLE_LATEST_20211027_030841_V17_7_0_120.SSA.pkg
    Finished list of software package changes
  [1] Finished Activate on R0
Checking status of Activate on [R0]
Activate: Passed on [R0]
Finished Activate

Send model notification for install_activate before reload
Install will reload the system now!
SUCCESS: install_activate  Fri Oct 29 00:05:09 UTC 2021

Router#
*Oct 29 00:05:09.504: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install activate PACKAGEOct 29 00:05:14.494: %PMAN-5-EXITACTION: R0/0: pvp: Process manager
 is exiting: reload action requested

Initializing Hardware ...

Checking for PCIe device presence...done
System integrity status: 0x610

System Bootstrap, Version 17.3(4.1r), RELEASE SOFTWARE
Copyright (c) 1994-2021 by cisco Systems, Inc.

Current image running   : Boot ROM1
Last reset cause        : LocalSoft
C8300-2N2S-6T platform with 8388608 Kbytes of main memory

□

Press RETURN to get started!

□

Router# install commit
install_commit: START Fri Oct 29 00:13:58 UTC 2021
install_commit: Committing PACKAGE

--- Starting Commit ---
Performing Commit on Active/Standby

*Oct 29 00:13:59.552: %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started install
 commit  [1] Commit package(s) on R0
  [1] Finished Commit on R0
Checking status of Commit on [R0]
Commit: Passed on [R0]
Finished Commit

SUCCESS: install_commit  Fri Oct 29 00:14:03 UTC 2021

Router#
*Oct 29 00:14:03.712: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install commit PACKAGE
```

The following is an example of downgrading in install mode:

```
ROUTER# install activate file bootflash:c8000be-universalk9.17.06.01a.SPA.bin activate
commit

install_add_activate_commit: START Fri Dec 10 18:07:17 GMT 2021

*Dec 10 18:07:18.405 GMT: %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
install one-shot bootflash:c8000be-universalk9.17.06.01a.SPA.bininstall_add_activate_commit:
 Adding PACKAGE
install_add_activate_commit: Checking whether new add is allowed ....

--- Starting Add ---
Performing Add on Active/Standby
  [1] Add package(s) on R0
  [1] Finished Add on R0
Checking status of Add on [R0]
Add: Passed on [R0]
Finished Add

Image added. Version: 17.06.01a.0.298
install_add_activate_commit: Activating PACKAGE
Following packages shall be activated:
/bootflash/c8000be-rpboot.17.06.01a.SPA.pkg
/bootflash/c8000be-mono-universalk9.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_sm_nim_adpt.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_sm_dsp_sp2700.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_sm_async.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_sm_1t3e3.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_sm_10g.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_prince.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_nim_xdsl.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_nim_ssd.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_nim_shdsl.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_nim_ge.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_nim_cwan.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_nim_bri_st_fw.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_nim_async.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_ngwic_t1e1.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_dsp_tilegx.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_dsp_sp2700.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_dsp_analogbri.17.06.01a.SPA.pkg
/bootflash/c8000be-firmware_dreamliner.17.06.01a.SPA.pkg

This operation may require a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
Performing Activate on Active/Standby
  [1] Activate package(s) on R0
  [1] Finished Activate on R0
Checking status of Activate on [R0]
Activate: Passed on [R0]
Finished Activate

--- Starting Commit ---
Performing Commit on Active/Standby
  [1] Commit package(s) on R0
Building configuration...

  [1] Finished Commit on R0
Checking status of Commit on [R0]
Commit: Passed on [R0]
Finished Commit

[OK]
*Dec 10 18:14:57.782 GMT: %SYS-6-PRIVCFG_ENCRYPT_SUCCESS: Successfully encrypted private
config fileSend model notification for install_add_activate_commit before reload
```

```
/usr/binos/conf/install_util.sh: line 164: /bootflash/.prst_sync/reload_info: No such file
 or directory
/usr/binos/conf/install_util.sh: line 168: /bootflash/.prst_sync/reload_info: No such file
 or directory
cat: /bootflash/.prst_sync/reload_info: No such file or directory
Install will reload the system now!
SUCCESS: install_add_activate_commit  Fri Dec 10 18:15:23 GMT 2021

ROUTER#
*Dec 10 18:15:23.955 GMT: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
 install one-shot PACKAGE bootflash:c8000be-universalk9.17.06.01a.SPA.binDec 10 18:15:27.708:
 %PMAN-5-EXITACTION: R0/0: pvp: Process manager is exiting: reload action requested

Initializing Hardware ...

Checking for PCIe device presence...done
System integrity status: 0x610
Rom image verified correctly

System Bootstrap, Version 17.3(5r), RELEASE SOFTWARE
Copyright (c) 1994-2021 by cisco Systems, Inc.

Current image running: Boot ROM0

Last reset cause: LocalSoft
ROUTER platform with 8388608 Kbytes of main memory

□

Press RETURN to get started!

□

ROUTER#
ROUTER# show version
Cisco IOS XE Software, Version 17.06.01a
Cisco IOS Software [Bengaluru], c8000be Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Version
 17.6.1a, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2021 by Cisco Systems, Inc.
Compiled Sat 21-Aug-21 03:27 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2021 by cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.

ROM: 17.3(5r)

ROUTER uptime is 0 minutes
Uptime for this control processor is 2 minutes
System returned to ROM by LocalSoft
System image file is "bootflash:packages.conf"
Last reload reason: LocalSoft

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
```

```
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

Technology Package License Information:

-----------------------------------------------------------------
Technology     Type         Technology-package Technology-package
                            Current            Next Reboot
-----------------------------------------------------------------
Smart License  Perpetual    None               None
Smart License  Subscription None               None

The current crypto throughput level is 250000 kbps

Smart Licensing Status: Registration Not Applicable/Not Applicable

cisco ROUTER (1RU) processor with 3747220K/6147K bytes of memory.
Processor board ID FDO2521M27S
Router operating mode: Autonomous
5 Gigabit Ethernet interfaces
2 2.5 Gigabit Ethernet interfaces
2 Cellular interfaces
32768K bytes of non-volatile configuration memory.
8388608K bytes of physical memory.
7573503K bytes of flash memory at bootflash:.
1875361792K bytes of NVMe SSD at harddisk:.
16789568K bytes of USB flash at usb0:.

Configuration register is 0x2102
```

The following is an example of terminating a software installation:

```
Router# install abort
install_abort: START Fri Oct 29 02:42:51 UTC 2021

This install abort would require a reload. Do you want to proceed? [y/n]
                                                                  *Oct 29 02:42:52.789:
 %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started install aborty
--- Starting Abort ---
Performing Abort on Active/Standby

  [1] Abort package(s) on R0
  [1] Finished Abort on R0
Checking status of Abort on [R0]
Abort: Passed on [R0]
Finished Abort

Send model notification for install_abort before reload
Install will reload the system now!
SUCCESS: install_abort  Fri Oct 29 02:44:47 UTC 2021

Router#
*Oct 29 02:44:47.866: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed
install abort PACKAGEOct 29 02:44:51.577: %PMAN-5-EXITACTION: R0/0: pvp: Process manager
```

```
is exiting: reload action requested

Initializing Hardware ...

Checking for PCIe device presence...done
System integrity status: 0x610

System Bootstrap, Version 17.3(4.1r), RELEASE SOFTWARE
Copyright (c) 1994-2021 by cisco Systems, Inc.

Current image running   : Boot ROM1
Last reset cause        : LocalSoft
C8300-2N2S-6T platform with 8388608 Kbytes of main memory

□

Press RETURN to get started!

□
```

The following are sample outputs for show commands:

show install log

```
Device# show install log
[0|install_op_boot]: START Thu Oct 28 22:09:29 Universal 2021
[0|install_op_boot(INFO, )]: Mount IMG INI state base image
[0|install_op_boot]: END SUCCESS  Thu Oct 28 22:09:30 Universal 2021
```

show install summary

```
Device# show install summary
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------
IMG   C    17.07.01.0.1515

--------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------
```

show install package *filesystem: filename*

```
Device# show install package
bootflash:c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.bin
Package: c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.bin
  Size: 831447859
  Timestamp: 2021-10-23 17:08:14 UTC
  Canonical path:
/bootflash/c8000be-universalk9.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.bin


  Raw disk-file SHA1sum:
    5c4e7617a6c71ffbcc73dcd034ab58bf76605e3f
  Header size:     1192 bytes
  Package type:    30000
  Package flags:   0
  Header version:  3

  Internal package information:
```

```
    Name: rp_super
    BuildTime: 2021-10-21_13.00
    ReleaseDate: 2021-10-21_03.11
    BootArchitecture: i686
    RouteProcessor: radium
    Platform: C8000BE
    User: mcpre
    PackageName: universalk9
    Build: BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117
    CardTypes:

  Package is bootable from media and tftp.
  Package contents:

  Package:
c8000be-firmware_nim_ge.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
    Size: 2966620
    Timestamp: 2021-10-21 20:10:44 UTC

    Raw disk-file SHA1sum:
      501d59d5f152ca00084a0da8217bf6f6b95dddb1
    Header size:     1116 bytes
    Package type:    40000
    Package flags:   0
    Header version:  3

    Internal package information:
      Name: firmware_nim_ge
      BuildTime: 2021-10-21_13.00
      ReleaseDate: 2021-10-21_03.11
      BootArchitecture: none
      RouteProcessor: radium
      Platform: C8000BE
      User: mcpre
      PackageName: firmware_nim_ge
      Build: BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117
      CardTypes:

    Package is not bootable.
  Package:
c8000be-firmware_prince.BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117.SSA.pkg
    Size: 10204252
    Timestamp: 2021-10-21 20:10:43 UTC

    Raw disk-file SHA1sum:
      a57bed4ddecfd08af3b456f69d11aaeb962865ea
    Header size:     1116 bytes
    Package type:    40000
    Package flags:   0
    Header version:  3

    Internal package information:
      Name: firmware_prince
      BuildTime: 2021-10-21_13.00
      ReleaseDate: 2021-10-21_03.11
      BootArchitecture: none
      RouteProcessor: radium
      Platform: C8000BE
      User: mcpre
      PackageName: firmware_prince
      Build: BLD_V177_THROTTLE_LATEST_20211021_031123_V17_7_0_117
      CardTypes:

    Package is not bootable.
```

show install active

```
Device# show install active
[ R0 ] Active Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------
IMG   C    17.07.01.0.1515

--------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------
```

show install inactive

```
Device# show install inactive
[ R0 ] Inactive Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------
No Inactive Packages
```

show install committed

```
Device# show install committed
[ R0 ] Committed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------
IMG   C    17.07.01.0.1515

--------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------
```

show install uncommitted

```
Device# show install uncommitted
[ R0 ] Uncommitted Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------
No Uncommitted Packages
```

Troubleshooting Software Installation Using install Commands

Problem Troubleshooting the software installation

Solution Use the following show commands to view installation summary, logs, and software versions.

- **show install summary**
- **show install log**

- **show version**
- **show version running**

Problem Other installation issues

Solution Use the following commands to resolve installation issue:

- **dir** <*install directory*>
- **more location:***packages.conf*
- **show tech-support install**: this command automatically runs the **show** commands that display information specific to installation.
- **request platform software trace archive target bootflash** <*location*>: this command archives all the trace logs relevant to all the processes running on the system since the last reload, and saves this information in the specified location.

CHAPTER 10

Support for Security-Enhanced Linux

This chapter describes the SELinux feature, and includes the following sections:

Overview

Security-Enhanced Linux (SELinux) is a solution composed of Linux kernel security module and system utilities to incorporate a strong, flexible Mandatory Access Control (MAC) architecture into Cisco IOS-XE platforms.

SELinux provides an enhanced mechanism to enforce the separation of information, based on confidentiality and integrity requirements, which addresses threats of tampering and bypassing of application security mechanisms and enables the confinement of damage that malicious or flawed applications can cause.

Prerequisites for SELinux

There are no specific prerequisites for this feature.

Restrictions for SELinux

There are no specific restrictions for this feature.

Information About SELinux

SELinux enforces mandatory access control policies that confine user programs and system services to the minimum privilege required to perform their assigned functionality. This reduces or eliminates the ability of these programs and daemons to cause harm when compromised (for example, through buffer overflows or misconfigurations). This is a practical implementation of principle of least privilege by enforcing MAC on Cisco IOS-XE platforms. This confinement mechanism works independently of the traditional Linux access control mechanisms. SELinux provides the capability to define policies to control the access from an application process to any resource object, thereby allowing for the clear definition and confinement of process behavior.

SELinux can operate either in **Permissive mode** or **Enforcing mode** when enabled on a system.

- In Permissive mode, SELinux does not enforce the policy, and only generates system logs for any denials caused by violation of the resource access policy. The operation is not denied, but only logged for resource access policy violation.
- In Enforcing mode, the SELinux policy is enabled and enforced. It denies resource access based on the access policy rules, and generates system logs.

From Cisco IOS XE 17.13.1a, SELinux is enabled in Enforcing mode by default on supported Cisco IOS XE platforms. In the Enforcing mode, any system resource access that does not have the necessary allow policy is treated as a violation, and the operation is denied. The violating operation fails when a denial occurs, and system logs are generated. In Enforcing mode, the solution works in access-violation prevention mode.

Supported Platforms

From Cisco IOS XE 17.13.1a, SELinux is enabled on the following platforms:

- Cisco 1000 Series Aggregation Services Routers
- Cisco 1000 Series Integrated Services Routers
- Cisco 4000 Series Integrated Services Routers
- Cisco Catalyst 8000v Edge Software
- Cisco Catalyst 8200 Series Edge Platforms
- Cisco Catalyst 8300 Series Edge Platforms
- Cisco Catalyst 8500 and 8500L Series Edge Platforms
- Cisco VG Series Gateways: VG400, VG410, VG420, and VG450
- Cisco 1100 Terminal Services Gateway

Configuring SELinux

The are no additional requirements or configuration steps needed to enable or use the SELinux feature in Enforcing mode.

The following commands are introduced as part of the SELinux feature:

set platform software selinux {default | enforcing | permissive}

platform security selinux {enforcing | permissive}

show platform software selinux

Note These new commands are implemented as **service internal** commands.

Configuring SELinux (EXEC Mode)

Use the **set platform software selinux** command to configure SELinux in EXEC mode.

The following example shows SELinux configuration in EXEC mode:

```
Device# set platform software selinux ?

default  Set SELinux mode to default
enforcing  Set SELinux mode to enforcing
permissive  Set SELinux mode to permissive
```

Configuring SELinux (CONFIG Mode)

Use the **platform security selinux** command to configure SELinux in configuration mode.

The following example shows SELinux configuration in CONFIG mode:

```
Device(config)# platform security selinux

enforcing  Set SELinux policy to Enforcing mode
permissive  Set SELinux policy to Permissive mode

Device(config)# platform security selinux permissive

Device(config)#
*Oct 20 21:52:45.155: %IOSXE-1-PLATFORM: R0/0:
SELINUX_MODE_PROG: Platform Selinux confinement mode downgraded to permissive!

Device(config)#
```

Examples for SELinux

The following example shows the output for changing the mode from Enforcing to Permissive:

```
"*Oct 20 21:44:03.609: %IOSXE-1-PLATFORM: R0/0:
SELINUX_MODE_PROG: Platform Selinux confinement mode downgraded to permissive!"
```

The following example shows the output for changing the mode from Permissive to Enforcing:

```
"*Oct 20 21:44:34.160: %IOSXE-1-PLATFORM: R0/0:
SELINUX_MODE_PROG: Platform Selinux confinement mode upgraded to enforcing!"
```

Note If the SELinux mode is changed, this change is considered a system security event, and a system log message is generated.

SysLog Message Reference

Facility-Severity-Mnemonic	**%SELINUX-1-VIOLATION**
Severity-Meaning	Alert Level Log
Message	N/A
Message Explanation	Resource access was made by the process for which a resource access policy does not exist. The operation was flagged, and resource access was denied. A system log was generated with information that process resource access has been denied.
Component	SELINUX
Recommended Action	Contact Cisco TAC with the following relevant information as attachments: • The exact message as it appears on the console or in the system • Output of the **show tech-support** command (text file) • Archive of Btrace files from the box using the following command: **request platform software trace archive target <URL>** • Output of the **show platform software selinux** command

The following examples demonstrate sample syslog messages:

Example 1:

```
*Nov 14 00:09:04.943: %SELINUX-1-VIOLATION: R0/0: audispd: type=AVC
msg=audit(1699927057.934:129): avc: denied { getattr } for pid=5899 comm="ls"
path="/root/test" dev="rootfs" ino=25839
scontext=system_u:system_r:polaris_iosd_t:s0
tcontext=system_u:object_r:admin_home_t:s0 tclass=file permissive=0
```

Example 2:

```
*Nov 14 00:09:04.947: %SELINUX-1-VIOLATION: R0/0: audispd: t type=AVC
msg=audit(1699927198.486:130): avc: denied { write } for pid=6012 comm="echo"
path="/root/test" dev="rootfs" ino=25839
scontext=system_u:system_r:polaris_iosd_t:s0
tcontext=system_u:object_r:admin_home_t:s0 tclass=file permissive= 0
```

Verifying SELinux Enablement

Use the **show platform software selinux** command to view the SELinux configuration mode:

```
Device# show platform software selinux
=========================================
IOS-XE SELINUX STATUS
=========================================
SElinux Status :    Enabled
Current Mode :      Enforcing
Config file Mode :  Enforcing
```

Troubleshooting SELinux

If there is an instance of an SELinux violation on your device or network, please reach out to Cisco TAC with the following details:

- The message exactly as it appears on the console or in the system log. For example:

  ```
  device#request platform software trace archive target
      flash:selinux_btrace_logs
  ```

- Output of the **show tech-support** command (text file)
- Archive of Btrace files from the box using the following command:

  request platform software trace archive target <URL>

- Output of the **show platform software selinux** command

CHAPTER 11

Slot and Subslot Configuration

This chapter contains information on slots and subslots. Slots specify the chassis slot number in your router and subslots specify the slot where the service modules are installed.

For further information on the slots and subslots, see the "About Slots and Interfaces" section in the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

The following section is included in this chapter:

Configuring the Interfaces

The following sections describe how to configure Gigabit interfaces and also provide examples of configuring the router interfaces:

Configuring Gigabit Ethernet Interfaces

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface GigabitEthernet** *slot/subslot/port*
4. **ip address** *ip-address mask* [**secondary**] **dhcp pool**
5. **negotiation auto**
6. **end**

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	**interface GigabitEthernet** *slot/subslot/port* **Example:** `Router(config)# interface GigabitEthernet 0/0/1`	Configures a GigabitEthernet interface. • **GigabitEthernet**—Type of interface. • *slot*—Chassis slot number. • */subslot*—Secondary slot number. The slash (/) is required. • /port—Port or interface number. The slash (/) is required.
Step 4	**ip address** *ip-address mask* [**secondary**] **dhcp pool** **Example:** `Router(config-if)# ip address 10.0.0.1 255.255.255.0 dhcp pool`	Assigns an IP address to the GigabitEthernet • **ip address** *ip-address*—IP address for the interface. • *mask*—Mask for the associated IP subnet. • **secondary** (optional)—Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address. • **dhcp**—IP address negotiated via DHCP. • **pool**—IP address autoconfigured from a local DHCP pool.
Step 5	**negotiation auto** **Example:** `Router(config-if)# negotiation auto`	Selects the negotiation mode. • **auto**—Performs link autonegotiation.
Step 6	**end** **Example:** `Router(config-if)# end`	Ends the current configuration session and returns to privileged EXEC mode.

Configuring the Interfaces: Example

The following example shows the **interface gigabitEthernet** command being used to add the interface and set the IP address. **0/0/0** is the slot/subslot/port. The ports are numbered 0 to 3.

```
Router# show running-config interface gigabitEthernet 0/0/0
Building configuration...
Current configuration : 71 bytes
!
interface gigabitEthernet0/0/0
no ip address
negotiation auto
end

Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface gigabitEthernet 0/0/0
```

Viewing a List of All Interfaces: Example

In this example, the **show platform software interface summary** and **show interfaces summary** commands are used to display all the interfaces:

```
Router# show platform software interface summary
  Interface                 IHQ  IQD  OHQ  OQD  RXBS  RXPS  TXBS  TXPS  TRTL
-----------------------------------------------------------------------------
* GigabitEthernet0/0/0        0    0    0    0     0     0     0     0     0
* GigabitEthernet0/0/1        0    0    0    0     0     0     0     0     0
* GigabitEthernet0/0/2        0    0    0    0     0     0     0     0     0
* GigabitEthernet0/0/3        0    0    0    0     0     0     0     0     0
* GigabitEthernet0            0    0    0    0     0     0     0     0     0


Router# show interfaces summary
   *: interface is up
 IHQ: pkts in input hold queue     IQD: pkts dropped from input queue
 OHQ: pkts in output hold queue    OQD: pkts dropped from output queue
 RXBS: rx rate (bits/sec)          RXPS: rx rate (pkts/sec)
 TXBS: tx rate (bits/sec)          TXPS: tx rate (pkts/sec)
 TRTL: throttle count

Interface             IHQ  IQD    OHQ    OQD    RXBS    RXPS    TXBS    TXPS    TRTL
------------------------------------------------------------------------------------------

* GigabitEthernet0/0/0 0    0      0      0      0       0       0       0       0
* GigabitEthernet0/0/1 0    0      0      0      0       0       0       0       0
* GigabitEthernet0/0/2 0    0      0      0      0       0       0       0       0
* GigabitEthernet0/0/3 0    0      0      0      0       0       0       0       0
* GigabitEthernet      0    0      0      0      0       0       0       0       0
```

Viewing Information About an Interface: Example

The following example shows how to display a brief summary of an interface's IP information and status, including the virtual interface bundle information, by using the **show ip interface brief** command:

```
Router# show ip interface brief
Interface              IP-Address      OK?  Method  Status                 Protocol
GigabitEthernet0/0/0    10.0.0.1       YES  manual  down                   down
GigabitEthernet0/0/1    unassigned     YES  NVRAM   administratively down  down
```

```
GigabitEthernet0/0/2    10.10.10.1     YES  NVRAM   up                      up
GigabitEthernet0/0/3    10.8.8.1       YES  NVRAM   up                      up
GigabitEthernet0        172.18.42.33   YES  NVRAM   up                      up
```

CHAPTER 12

Cisco Thousand Eyes Enterprise Agent Application Hosting

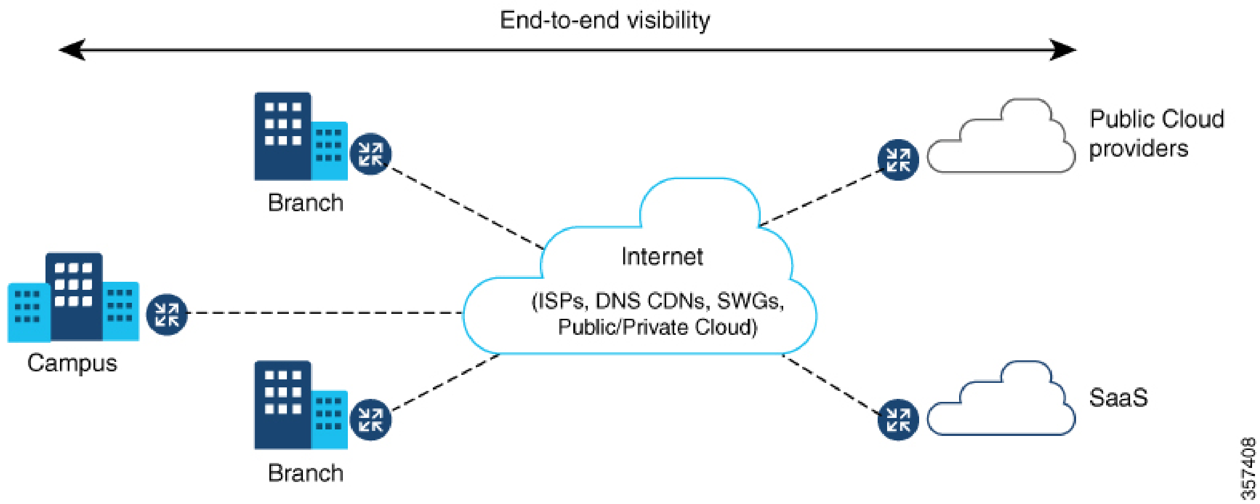
This chapter provides information on Cisco Thousand Eyes Enterprise Agent Application Hosting. The following sections are included in this chapter:

Cisco ThousandEyes Enterprise Agent Application Hosting

Cisco ThousandEyes is a network intelligence platform that allows you to use its agents to run a variety of tests from its agents to monitor the network and application performance. This application enables you to view end-to-end paths across networks and services that impact your business. Cisco ThousandEyes application actively monitors the network traffic paths across internal, external, and internet networks in real time, and helps to analyse the network performance. Also, Cisco ThousandEyes application provides application availability insights that are enriched with routing and device data for a multidimensional view of digital experience.

From Cisco IOS XE Release 17.6.1, you can use application hosting capabilities to deploy the Cisco ThousandEyes Enterprise Agent as a container application on Cisco 4000 Series Integrated Services Routers (ISRs). This agent application runs as a docker image using Cisco IOx docker-type option. For more information on how to configure Cisco ThousandEyes in controller mode, see Cisco SD-WAN Systems and Interfaces Configuration Guide.

Figure 1: Network View through ThousandEyes Application



Feature Information for Cisco ThousandEyes Enterprise Agent Application Hosting

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 20: Feature Information for ThousandEyes Enterprise Agent Application Hosting

Feature Name	Releases	Feature Information
Cisco ThousandEyes Enterprise Agent Application Hosting	Cisco IOS XE 17.7.1a	The Cisco ThousandEyes Enterprise Agent Application introduces the functionality to inherit the Domain Name Server (DNS) information from the device. With this enhancement, the DNS field in vManage ThousandEyes feature template is an optional parameter.
Cisco ThousandEyes Enterprise Agent Application Hosting	Cisco IOS XE 17.6.1	With the integration of ThousandEyes Agent Application running on routing platforms using the app-hosting capabilities as container, you can have visibility into application experience with deep insights into the internet, cloud providers, and enterprise networks.

Supported Platforms and System Requirements

The following table lists the supported platforms and system requirements.

Table 21: Supported Platforms and System Requirements

Platforms	Bootflash	FRU Storage	DRAM
Cisco ISR 4000 Series			
ISR446x	8 GB	NIM-SSD (external)	8 GB, 16 GB, 32 GB
ISR4451	8 GB	NIM-SSD (external)	8 GB, 16 GB
ISR4351/31	16 GB	NIM-SSD (external)	8 GB, 16 GB
ISR4321	8 GB	NIM-SSD (external)	8 GB
ISR4221X	8 GB	NIM-SSD (external)	8 GB

Note The minimum DRAM and storage requirement for running Cisco ThousandEyes Enterprise Agent is 8 GB. If the device does not have enough memory or storage, we recommend that you upgrade DRAM or add an external storage such as M.2 USB. When the available resources are not sufficient to run other applications, Cisco IOx generates an error message.

Workflow to Install and Run the Cisco ThousandEyes Application

To install and run the Cisco ThousandEyes image on a device, perform these steps:

Step 1 Create a new account on the Cisco ThousandEyes portal.

Step 2 Download the Cisco ThousandEyes application package from the software downloads page and ensure that you use the agent version 4.0.2.

Step 3 Copy the image on the device.

Step 4 Install and launch the image.

Step 5 Connect the agent to the controller.

Note When you order platforms that support Cisco ThousandEyes application with Cisco IOS XE 17.6.1 software, the Cisco ThousandEyes application package is available in the bootflash of the device.

Workflow to Host the Cisco ThousandEyes Application

To install and launch the application, perform these steps:

Before you begin

Create a new account on the Cisco ThousandEyes portal and generate the token. The Cisco ThousandEyes agent application uses this token to authenticate and check into the correct Cisco ThousandEyes account. If you see a message stating that your token is invalid and you want to troubleshoot the issue, see Troubleshooting the Cisco ThousandEyes Application, on page 181 secion.

Note If you configure the correct token and Domain Name Server (DNS) information, the device is discovered automatically.

Step 1 Enable Cisco IOX application environment on the device.

- Use the following commands for non-SD-WAN (autonomous mode) images:

```
config terminal
 iox
end
write
```

- Use the following commands for SD-WAN (controller mode) images:

```
config-transaction
iox
commit
```

Step 2 If the IOx command is accepted, wait for a few seconds and check whether the IOx process is up and running by using the **show iox** command. The output must display that the show IOxman process is running.

```
Device #show iox

IOx Infrastructure Summary:
---------------------------
IOx service (CAF) 10.11.0.0     : Running
IOx service (HA)               : Not Supported
IOx service (IOxman)           : Running
IOx service (Sec storage)      : Not Supported
Libvirtd 1.3.4                 : Running
```

Step 3 Ensure that the ThousandEyes application LXC tarball is available in the device *bootflash:*.

Step 4 Create a virtual port group interface to enable the traffic path to the Cisco ThousandEyes application:

```
interface VirtualPortGroup 0
          ip address 192.168.35.1 255.255.255.0
        exit
```

Step 5 Configure the app-hosting application with the generated token:

```
app-hosting appid te
          app-vnic gateway1 virtualportgroup 0 guest-interface 0
          guest-ipaddress 192.168.35.2 netmask 255.255.255.0
          app-default-gateway 192.168.35.1 guest-interface 0
          app-resource docker
                 prepend-pkg-opts  Required to get the default run-time options from package.yaml
```

```
                   run-opts 1 "--hostname thousandeyes"
             run-opts 2 "-e TEAGENT_ACCOUNT_TOKEN=<ThousandEyes token>"
        run-opts 3 "-e TEAGENT_PROXY_TYPE=STATIC -e TEAGENT_PROXY_LOCATION=proxy.something.other:80"

          name-server0 10.75.75.75 □ ISP's DNS server
        end

app-hosting appid te
 app-resource docker
  prepend-pkg-opts
  run-opts 2 "--hostname
```

Note You can use the proxy configuration only if the Cisco ThousandEyes agent does not have an internet access without a proxy. Also, the hostname is optional. If you do not provide the hostname during the installation, the device hostname is used as the Cisco ThousandEyes agent hostname. The device hostname is displayed on the Cisco ThousandEyes portal. The DNS name server information is optional. If the Cisco ThousandEyes agent uses a private IP address, ensure that you establish a connection to the device through NAT.

Step 6 Configure the **start** command to run the application automatically when the application is installed on the device using the **install** command:

```
app-hosting appid te
         start
```

Step 7 Install the ThousandEyes application:

```
app-hosting install appid <appid> package [bootflash: | harddisk: | https:]
```

Select a location to install the ThousandEyes application from these options:

```
Device# app-hosting install appid te package ?
           bootflash:  Package path □ ISR4K case if image is locally available in bootflash:
           harddisk:   Package path □ Cat8K case if image is locally available in M.2 USB
           https:      Package path □ Download over the internet if image is not locally present in
 router. URL to ThousandEyes site hosting agent image to be provided here
```

Step 8 Check if the application is up and running:

```
Device#show app-hosting list
 App id                                   State
---------------------------------------------------------
  te                                       RUNNING
```

Note If any of these steps fail, use the **show logging** command and check the IOx error message. If the error message is about insufficient disk space, clean the storage media (bootflash or hard disk) to free up the space. Use the **show app-hosting resource** command to check the CPU and disk memory.

Downloading and Copying the Image to the Device

To download and copy the image to bootflash, perform these steps:

Step 1 Check if the Cisco ThousandEyes image is precopied to *bootflash:/<directory name>*.

Step 2 If the image is not available in the device directory, perform these steps:

a) If the device has a direct access to internet, use the *https:*. option in the **application install** command. This option downloads the image from the Cisco ThousandEyes software downloads page into *bootflash:/apps* and installs the application.

```
Device# app-hosting install appid <appid string> package [bootflash: | flash | http | https://
| ftp | ] URL to image location hosted on ThousandEyes portal

Device# app-hosting install appid te1000 package
https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-4.0.2.cisco.tar

Installing package
'https://downloads.thousandeyes.com/enterprise-agent/thousandeyes-enterprise-agent-4.0.2.cisco.tar'
 for 'te1000'.

Use 'show app-hosting list' for progress.
*Jun 29 23:43:29.244: %IOSXE-6-PLATFORM: R0/0: IOx:  App verification successful
*Jun 29 23:45:00.449: %IM-6-INSTALL_MSG: R0/0: ioxman: app-hosting: Install succeeded: te1000
installed successfully Current state is DEPLOYED
*Jun 29 23:45:01.801: %IOSXE-6-PLATFORM: R0/0: IOx:  App verification successful
*Jun 29 23:45:51.054: %IM-6-START_MSG: R0/0: ioxman: app-hosting: Start succeeded: te1000 started
 successfully Current state is RUNNING

Device#show app-hosting detail appid te1000  ( Details of Application)
App id                 : te1000
Owner                  : iox
State                  : RUNNING
Application
  Type                 : docker
  Name                 : ThousandEyes Enterprise Agent
  Version              : 4.0
  Author               : ThousandEyes <support@thousandeyes.com>
  Path                 : bootflash:thousandeyes-enterprise-agent-4.0-22.cisco.tar
Resource reservation
  Memory               : 500 MB
  Disk                 : 1 MB
  CPU                  : 1500 units
  CPU-percent          : 70 %
```

b) If the device has a proxy server, copy the image manually to *bootflash:/apps*.

c) Download the Cisco ThousandEyes application package from the software downloads page and ensure that you use the agent version 4.0.2.

d) Create an application directory in the *bootflash:* to copy the image:

```
Device# mkdir bootflash:apps
Create directory filename [apps]?
Created dir bootflash:/apps
```

e) Copy the Cisco ThousandEyes image to the *bootflash:apps* directory.

f) Validate the image using the **verify** command:

```
verify /md5 bootflash:apps/<file name>
```

Connecting the Cisco ThousandEyes Agent with the Controller

Before you begin

Ensure that you have an Internet connection before you connect the agent with the controller.

After the Cisco ThousandEyes application is up and running, the agent (ThousandEyes-agent) process connects to the controller that is running on the cloud environment.

Note If you have issues related to connectivity, the application logs the relevant error messages in the application-specific logs (*/var/logs*).

Modifying the Agent Parameters

To modify the agent parameters, perform these actions:

Step 1 Stop the application using the **app-hosting stop appid appid** command.

Step 2 Deactivate the application using the **app-hosting deactivate appid appid** command.

Step 3 Make the required changes to the app-hosting configuration.

Step 4 Activate the application using the **app-hosting activate appid appid** command.

Step 5 Start the application using the **app-hosting start appid appid** command.

Uninstalling the Application

To uninstall the application, perform these steps:

Step 1 Stop the application using the **app-hosting stop appid te** command.

Step 2 Check if the application is in active state using the **show app-hosting list** command.

Step 3 Deactivate the application using the **app-hosting deactivate appid te** command.

Step 4 Ensure that the application is not in active state. Use the **show app-hosting list** command to check status of the application.

Step 5 Uninstall the application using the **app-hosting uninstall appid te** command.

Step 6 After the uninstallation process is complete, use the **show app-hosting list** command to check if the application is uninstalled successfully.

Troubleshooting the Cisco ThousandEyes Application

To troubleshoot the Cisco ThousandEyes application, perform these steps:

1. Connect to Cisco ThousandEyes agent application using the **app-hosting connect appid appid session /bin/bash** command.

2. Verify the configuration applied to the application in */etc/te-agent.cfg*.

3. View the logs in */var/log/agent/te-agent.log*. You can use these logs to troubleshoot the configuration.

Checking the ThousandEyes Application Status

When the Cisco ThousandEyes application is in running state, it is registered on the ThousandEyes portal. If the application does not show up in a few minutes after the agent is in running state, check the following using the **app-hosting connect appid thousandeyes_enterprise_agent session** command:

```
Device#app-hosting connect appid thousandeyes_enterprise_agent session
Device# cat /var/log/agent/te-agent.log
2021-02-04 08:59:29.642 DEBUG [e4736a40] [te.agent.AptPackageInterface] {} Initialized APT
 package interface
2021-02-04 08:59:29.642 INFO  [e4736a40] [te.agent.main] {} Agent version 1.103.0 starting.
  Max core size is 0 and max open files is 1024
2021-02-04 08:59:29.642 DEBUG [e4736a40] [te.agent.db] {} Vacuuming database
2021-02-04 08:59:29.643 INFO  [e4736a40] [te.agent.db] {} Found version 0, expected version
 50
2021-02-04 08:59:29.672 INFO  [e4708700] [te.probe.ServerTaskExecutor] {} ProbeTaskExecutor
 started with 2 threads.
2021-02-04 08:59:29.673 INFO  [e2f05700] [te.probe.ProbeTaskExecutor.bandwidth] {}
ProbeTaskExecutor started with 1 threads.
2021-02-04 08:59:29.673 INFO  [e2704700] [te.probe.ProbeTaskExecutor.realtime] {}
ProbeTaskExecutor started with 1 threads.
2021-02-04 08:59:29.673 INFO  [e1f03700] [te.probe.ProbeTaskExecutor.throughput] {}
ProbeTaskExecutor started with 1 threads.
2021-02-04 08:59:29.674 DEBUG [e4736a40] [te.agent.DnssecTaskProceessor] {} Agent is not
running bind
2021-02-04 08:59:29.674 DEBUG [e4736a40] [te.snmp.RequestDispatcher] {} Initialised SNMP++
 session
2021-02-04 08:59:29.674 DEBUG [e4736a40] [te.snmp.RequestDispatcher] {} Initialised SNMP++
 session
2021-02-04 08:59:29.674 DEBUG [e4736a40] [te.snmp.RequestDispatcher] {} Initialised SNMP++
 session
2021-02-04 08:59:29.674 INFO  [e4736a40] [te.agent.main] {} Agent starting up
2021-02-04 08:59:29.675 INFO  [e4736a40] [te.agent.main] {} No agent id found, attempting
to obtain one
2021-02-04 08:59:29.675 INFO  [e4736a40] [te.agent.ClusterMasterAdapter] {} Attempting to
get agent id from sc1.thousandeyes.com
2021-02-04 08:59:29.679 ERROR [e4736a40] [te.agent.main] {} Error calling create_agent:
Curl error - Couldn't resolve host name
2021-02-04 08:59:29.680 INFO  [e4736a40] [te.agent.main] {} Sleeping for 30 seconds
Note :
```

Note Check the DNS server connection. If the Cisco ThousandEyes agent is assigned to a private IP address, check the NAT configuration.

CHAPTER 13

Process Health Monitoring

This chapter describes how to manage and monitor the health of various components of your router. It contains the following sections:

Monitoring Control Plane Resources

The following sections explain the of memory and CPU monitoring from the perspective of the Cisco IOS process and the overall control plane:

Avoiding Problems Through Regular Monitoring

Processes should provide monitoring and notification of their status/health to ensure correct operation. When a process fails, a syslog error message is displayed and either the process is restarted or the router is rebooted. A syslog error message is displayed when a monitor detects that a process is stuck or has crashed. If the process can be restarted, it is restarted; else, the router is restarted.

Monitoring system resources enables you to detect potential problems before they occur, thus avoiding outages. The following are the advantages of regular monitoring:

- Lack of memory on line cards that are in operation for a few years can lead to major outages. Monitoring memory usage helps to identify memory issues in the line cards and enables you to prevent an outage.
- Regular monitoring establishes a baseline for a normal system load. You can use this information as a basis for comparison when you upgrade hardware or software—to see if the upgrade has affected resource usage.

Cisco IOS Process Resources

You can view CPU utilization statistics on active processes and see the amount of memory being used in these processes using the **show memory** command and the **show process cpu** command. These commands provide a representation of memory and CPU utilization from the perspective of only the Cisco IOS process; they do not include information for resources on the entire platform. For example, when the **show memory** command is used in a system with 8 GB RAM running a single Cisco IOS process, the following memory usage is displayed:

```
Router# show memory
           Head           Total(b)     Used(b)     Free(b)      Lowest(b)    Largest(b)
Processor  2ABEA4316010   4489061884   314474916   4174586968   3580216380   3512323496
lsmpi_io   2ABFAFF471A8   6295128      6294212     916          916          916
Critical   2ABEB7C72EB0   1024004      92          1023912      1023912      1023912
```

The **show process cpu** command displays Cisco IOS CPU utilization average:

```
Router# show process cpu
CPU utilization for five seconds: 0%/0%; one minute: 0%; five minutes: 0%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
   1         583     48054         12  0.00%  0.00%  0.00%   0 Chunk Manager
   2         991    176805          5  0.00%  0.00%  0.00%   0 Load Meter
   3           0         2          0  0.00%  0.00%  0.00%   0 IFCOM Msg Hdlr
   4           0        11          0  0.00%  0.00%  0.00%   0 Retransmission o
   5           0         3          0  0.00%  0.00%  0.00%   0 IPC ISSU Dispatc
   6      230385    119697       1924  0.00%  0.01%  0.00%   0 Check heaps
   7          49        28       1750  0.00%  0.00%  0.00%   0 Pool Manager
   8           0         2          0  0.00%  0.00%  0.00%   0 Timers
   9       17268    644656         26  0.00%  0.00%  0.00%   0 ARP Input
  10         197    922201          0  0.00%  0.00%  0.00%   0 ARP Background
  11           0         2          0  0.00%  0.00%  0.00%   0 ATM Idle Timer
  12           0         1          0  0.00%  0.00%  0.00%   0 ATM ASYNC PROC
  13           0         1          0  0.00%  0.00%  0.00%   0 AAA_SERVER_DEADT
  14           0         1          0  0.00%  0.00%  0.00%   0 Policy Manager
  15           0         2          0  0.00%  0.00%  0.00%   0 DDR Timers
  16           1        15         66  0.00%  0.00%  0.00%   0 Entity MIB API
  17          13      1195         10  0.00%  0.00%  0.00%   0 EEM ED Syslog
  18          93        46       2021  0.00%  0.00%  0.00%   0 PrstVbl
  19           0         1          0  0.00%  0.00%  0.00%   0 RO Notify Timers
```

Overall Control Plane Resources

Control plane memory and CPU utilization on each control processor allows you to keep a tab on the overall control plane resources. You can use the **show platform software status control-processor brief** command (summary view) or the **show platform software status control-processor** command (detailed view) to view control plane memory and CPU utilization information.

All control processors should show status, Healthy. Other possible status values are Warning and Critical. Warning indicates that the router is operational, but that the operating level should be reviewed. Critical implies that the router is nearing failure.

If you see a Warning or Critical status, take the following actions:

- Reduce the static and dynamic loads on the system by reducing the number of elements in the configuration or by limiting the capacity for dynamic services.
- Reduce the number of routes and adjacencies, limit the number of ACLs and other rules, reduce the number of VLANs, and so on.

The following sections describe the fields in the **show platform software status control-processor** command output.

Load Average

Load average represents the process queue or process contention for CPU resources. For example, on a single-core processor, an instantaneous load of 7 would mean that seven processes are ready to run, one of which is currently running. On a dual-core processor, a load of 7 would mean that seven processes are ready to run, two of which are currently running.

Memory Utilization

Memory utilization is represented by the following fields:

- Total—Total line card memory
- Used—Consumed memory
- Free—Available memory
- Committed—Virtual memory committed to processes

CPU Utilization

CPU utilization is an indication of the percentage of time the CPU is busy, and is represented by the following fields:

- CPU—Allocated processor
- User—Non-Linux kernel processes
- System—Linux kernel process
- Nice—Low-priority processes
- Idle—Percentage of time the CPU was inactive
- IRQ—Interrupts
- SIRQ—System Interrupts
- IOwait—Percentage of time CPU was waiting for I/O

Example: show platform software status control-processor Command

The following are some examples of using the **show platform software status control-processor** command:

```
Router# show platform software status control-processor
RP0: online, statistics updated 5 seconds ago
Load Average: healthy
  1-Min: 0.07, status: healthy, under 5.00
  5-Min: 0.11, status: healthy, under 5.00
  15-Min: 0.09, status: healthy, under 5.00
Memory (kb): healthy
  Total: 3971216
  Used: 3415976 (86%)
  Free: 555240 (14%)
```

```
  Committed: 2594412 (65%), status: healthy, under 90%
Per-core Statistics
CPU0: CPU Utilization (percentage of time spent)
  User:  1.40, System:  1.20, Nice:  0.00, Idle: 97.39
  IRQ:  0.00, SIRQ:  0.00, IOwait:  0.00
CPU1: CPU Utilization (percentage of time spent)
  User:  0.89, System:  0.79, Nice:  0.00, Idle: 98.30
  IRQ:  0.00, SIRQ:  0.00, IOwait:  0.00
CPU2: CPU Utilization (percentage of time spent)
  User:  0.80, System:  2.50, Nice:  0.00, Idle: 96.70
  IRQ:  0.00, SIRQ:  0.00, IOwait:  0.00
CPU3: CPU Utilization (percentage of time spent)
  User:  3.09, System:  6.19, Nice:  0.00, Idle: 90.60
  IRQ:  0.00, SIRQ:  0.09, IOwait:  0.00
CPU4: CPU Utilization (percentage of time spent)
  User:  0.10, System:  0.30, Nice:  0.00, Idle: 99.60
  IRQ:  0.00, SIRQ:  0.00, IOwait:  0.00
CPU5: CPU Utilization (percentage of time spent)
  User:  0.89, System:  1.59, Nice:  0.00, Idle: 97.50
  IRQ:  0.00, SIRQ:  0.00, IOwait:  0.00
CPU6: CPU Utilization (percentage of time spent)
  User:  0.80, System:  1.10, Nice:  0.00, Idle: 98.10
  IRQ:  0.00, SIRQ:  0.00, IOwait:  0.00
CPU7: CPU Utilization (percentage of time spent)
  User:  0.20, System:  3.40, Nice:  0.00, Idle: 96.40
  IRQ:  0.00, SIRQ:  0.00, IOwait:  0.00

Router# show platform software status control-processor brief
Load Average
 Slot  Status  1-Min  5-Min 15-Min
  RP0 Healthy   0.09   0.10   0.09

Memory (kB)
 Slot  Status    Total     Used (Pct)     Free (Pct) Committed (Pct)
  RP0 Healthy  3971216  3426452 (86%)   544764 (14%)   2595212 (65%)

CPU Utilization
 Slot  CPU   User System   Nice   Idle    IRQ   SIRQ IOwait
  RP0    0   1.60   0.90   0.00  97.30   0.10   0.10   0.00
         1   0.09   1.29   0.00  98.60   0.00   0.00   0.00
         2   0.10   0.10   0.00  99.79   0.00   0.00   0.00
         3   0.00   0.00   0.00 100.00   0.00   0.00   0.00
         4   0.60   4.90   0.00  94.50   0.00   0.00   0.00
         5   0.70   1.30   0.00  98.00   0.00   0.00   0.00
         6   0.10   0.00   0.00  99.90   0.00   0.00   0.00
         7   1.39   0.49   0.00  98.10   0.00   0.00   0.00
```

Monitoring Hardware Using Alarms

- Router Design and Monitoring Hardware, on page 187
- BootFlash Disk Monitoring, on page 187
- Approaches for Monitoring Hardware Alarms, on page 187

Router Design and Monitoring Hardware

The router sends alarm notifications when problems are detected, allowing you to monitor the network remotely. You do not need to use **show** commands to poll devices on a routine basis; however, you can perform onsite monitoring if you choose.

BootFlash Disk Monitoring

The bootflash disk must have enough free space to store two core dumps. This condition is monitored, and if the bootflash disk is too small to store two core dumps, a syslog alarm is generated, as shown in the following example:

```
Aug 22 13:40:41.038 R0/0: %FLASH_CHECK-3-DISK_QUOTA: Flash disk quota exceeded
[free space is 7084440 kB] - Please clean up files on bootflash.
```

The size of the bootflash disk must be at least of the same size as that of the physical memory installed on the router. If this condition is not met, a syslog alarm is generated as shown in the following example:

```
%IOSXEBOOT-2-FLASH_SIZE_CHECK: (rp/0): Flash capacity (8 GB) is insufficient for fault
analysis based on
installed memory of RP (16 GB)
%IOSXEBOOT-2-FLASH_SIZE_CHECK: (rp/0): Please increase the size of installed flash to at
least 16 GB (same as
physical memory size)
```

Approaches for Monitoring Hardware Alarms

- Onsite Network Administrator Responds to Audible or Visual Alarms, on page 187
- Viewing the Console or Syslog for Alarm Messages, on page 188
- Network Management System Alerts a Network Administrator when an Alarm is Reported Through SNMP, on page 190

Onsite Network Administrator Responds to Audible or Visual Alarms

- About Audible and Visual Alarms, on page 187
- Clearing an Audible Alarm, on page 187
- Clearing a Visual Alarm, on page 188

About Audible and Visual Alarms

An external element can be connected to a power supply using the DB-25 alarm connector on the power supply. The external element is a DC light bulb for a visual alarm and a bell for an audible alarm.

If an alarm illuminates the CRIT, MIN, or MAJ LED on the faceplate of the router, and a visual or audible alarm is wired, the alarm also activates an alarm relay in the power supply DB-25 connector, and either the bell rings or the light bulb flashes.

Clearing an Audible Alarm

To clear an audible alarm, perform one of the following tasks:

- Press the **Audible Cut Off** button on the faceplate.
- Enter the **clear facility-alarm** command.

Clearing a Visual Alarm

To clear a visual alarm, you must resolve the alarm condition. The **clear facility-alarm** command does not clear an alarm LED on the faceplate or turn off the DC light bulb. For example, if a critical alarm LED is illuminated because an active module was removed without a graceful deactivation, the only way to resolve that alarm is to replace the module.

Viewing the Console or Syslog for Alarm Messages

The network administrator can monitor alarm messages by reviewing alarm messages sent to the system console or to a system message log (syslog).

- Enabling the logging alarm Command, on page 188
- Examples of Alarm Messages, on page 188
- Reviewing and Analyzing Alarm Messages, on page 190

Enabling the logging alarm Command

The **logging alarm** command must be enabled for the system to send alarm messages to a logging device, such as the console or a syslog. This command is not enabled by default.

You can specify the severity level of the alarms to be logged. All the alarms at and above the specified threshold generate alarm messages. For example, the following command sends only critical alarm messages to logging devices:

```
Router(config)# logging alarm critical
```

If alarm severity is not specified, alarm messages for all severity levels are sent to logging devices.

Examples of Alarm Messages

The following are examples of alarm messages that are sent to the console when a module is removed before performing a graceful deactivation. The alarm is cleared when the module is reinserted.

Module Removed

```
*Aug 22 13:27:33.774: %ISR4451-X_OIR-6-REMSPA: Module removed from subslot 1/1, interfaces
 disabled
*Aug 22 13:27:33.775: %SPA_OIR-6-OFFLINECARD: Module (SPA-4XT-SERIAL) offline in subslot
1/1
```

Module Reinserted

```
*Aug 22 13:32:29.447: %ISR4451-X_OIR-6-INSSPA: Module inserted in subslot 1/1
*Aug 22 13:32:34.916: %SPA_OIR-6-ONLINECARD: Module (SPA-4XT-SERIAL) online in subslot 1/1
*Aug 22 13:32:35.523: %LINK-3-UPDOWN: SIP1/1: Interface EOBC1/1, changed state to up
```

Alarms

To view alarms, use the **show facility-alarm status** command. The following example shows a critical alarm for the power supply:

```
Router# show facility-alarm status
System Totals  Critical: 5  Major: 0  Minor: 0

Source                     Severity      Description [Index]
------                     --------      -------------------
Power Supply Bay 0         CRITICAL      Power Supply/FAN Module Missing [0]
GigabitEthernet0/0/0       INFO          Physical Port Link Down [1]
GigabitEthernet0/0/1       INFO          Physical Port Link Down [1]
GigabitEthernet0/0/2       INFO          Physical Port Link Down [1]
GigabitEthernet0/0/3       INFO          Physical Port Link Down [1]
xcvr container 0/0/0       INFO          Transceiver Missing [0]
xcvr container 0/0/1       INFO          Transceiver Missing [0]
xcvr container 0/0/2       INFO          Transceiver Missing [0]
xcvr container 0/0/3       INFO          Transceiver Missing [0]
```

To view critical alarms, use the **show facility-alarm status critical** command, as shown in the following example:

```
Router# show facility-alarm status critical
System Totals  Critical: 5  Major: 0  Minor: 0

Source                     Severity      Description [Index]
------                     --------      -------------------
Power Supply Bay 0         CRITICAL      Power Supply/FAN Module Missing [0]
GigabitEthernet0/0/0       INFO          Physical Port Link Down [1]
GigabitEthernet0/0/1       INFO          Physical Port Link Down [1]
GigabitEthernet0/0/2       INFO          Physical Port Link Down [1]
GigabitEthernet0/0/3       INFO          Physical Port Link Down [1]
```

To view the operational state of the major hardware components on the router, use the **show platform diag** command. This example shows that power supply P0 has failed:

```
Router# show platform diag
Chassis type: ISR4451/K9

Slot: 0, ISR4451-NGSM
  Running state              : ok
  Internal state             : online
  Internal operational state  : ok
  Physical insert detect time : 00:01:09 (1w0d ago)
  Software declared up time   : 00:01:42 (1w0d ago)
  CPLD version               : 12061320
  Firmware version           : 12.2(20120618:163328)[ciscouser-ESGROM_20120618_GAMMA 101]

Sub-slot: 0/0, ISR4451-4X1GE
  Operational status         : ok
  Internal state             : inserted
  Physical insert detect time : 00:02:48 (1w0d ago)
  Logical insert detect time  : 00:02:48 (1w0d ago)

Slot: 1, ISR4451-NGSM
  Running state              : ok
  Internal state             : online
  Internal operational state  : ok
  Physical insert detect time : 00:01:09 (1w0d ago)
  Software declared up time   : 00:01:43 (1w0d ago)
  CPLD version               : 12061320
  Firmware version           : 12.2(20120618:163328)[ciscouser-ESGROM_20120618_GAMMA 101]

Slot: 2, ISR4451-NGSM
  Running state              : ok
  Internal state             : online
  Internal operational state  : ok
```

```
  Physical insert detect time : 00:01:09 (1w0d ago)
  Software declared up time   : 00:01:44 (1w0d ago)
  CPLD version                : 12061320
  Firmware version            : 12.2(20120618:163328)[ciscouser-ESGROM_20120618_GAMMA 101]

Slot: R0, ISR4451/K9
  Running state               : ok, active
  Internal state              : online
  Internal operational state  : ok
  Physical insert detect time : 00:01:09 (1w0d ago)
  Software declared up time   : 00:01:09 (1w0d ago)
  CPLD version                : 12061320
  Firmware version            : 12.2(20120618:163328)[ciscouser-ESGROM_20120618_GAMMA 101]

Slot: F0, ISR4451-FP
  Running state               : init, active
  Internal state              : online
  Internal operational state  : ok
  Physical insert detect time : 00:01:09 (1w0d ago)
  Software declared up time   : 00:01:37 (1w0d ago)
  Hardware ready signal time  : 00:00:00 (never ago)
  Packet ready signal time    : 00:00:00 (never ago)
  CPLD version                :
  Firmware version            : 12.2(20120618:163328)[ciscouser-ESGROM_20120618_GAMMA 101]

Slot: P0, Unknown
  State                       : ps, fail
  Physical insert detect time : 00:00:00 (never ago)

Slot: P1, XXX-XXXX-XX
  State                       : ok
  Physical insert detect time : 00:01:26 (1w0d ago)

Slot: P2, ACS-4450-FANASSY
  State                       : ok
  Physical insert detect time : 00:01:26 (1w0d ago)
```

Reviewing and Analyzing Alarm Messages

To facilitate the review of alarm messages, you can write scripts to analyze alarm messages sent to the console or syslog. Scripts can provide reports on events such as alarms, security alerts, and interface status.

Syslog messages can also be accessed through Simple Network Management Protocol (SNMP) using the history table defined in the CISCO-SYSLOG-MIB.

Network Management System Alerts a Network Administrator when an Alarm is Reported Through SNMP

The SNMP is an application-layer protocol that provides a standardized framework and a common language used for monitoring and managing devices in a network. Of all the approaches to monitor alarms, SNMP is the best approach to monitor more than one router in an enterprise and service provider setup.

SNMP provides notification of faults, alarms, and conditions that might affect services. It allows a network administrator to access router information through a network management system (NMS) instead of reviewing logs, polling devices, or reviewing log reports.

To use SNMP to get alarm notification, use the following MIBs:

- ENTITY-MIB, RFC 4133 (required for the CISCO-ENTITY-ALARM-MIB and CISCO-ENTITY-SENSOR-MIB to work)

- CISCO-ENTITY-ALARM-MIB
- CISCO-ENTITY-SENSOR-MIB (for transceiver environmental alarm information, which is not provided through the CISCO-ENTITY-ALARM-MIB)

CHAPTER 14

System Messages

System messages are saved in a log file or directed to other devices from the software running on a router. These messages are also known as syslog messages. System messages provide you with logging information for monitoring and troubleshooting purposes.

The following sections are included in this chapter:

Information About Process Management

You can access system messages by logging in to the console through Telnet protocol and monitoring your system components remotely from any workstation that supports the Telnet protocol.

Starting and monitoring software is referred to as process management. The process management infrastructure for a router is platform independent, and error messages are consistent across platforms running on Cisco IOS XE. You do not have to be directly involved in process management, but we recommend that you read the system messages that refer to process failures and other issues.

How to Find Error Message Details

To show further details about a process management or a syslog error message, enter the error message into the Error Message Decoder tool at: https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi.

For example, enter the message `%PMAN-0-PROCESS_NOTIFICATION` into the tool to view an explanation of the error message and the recommended action to be taken.

The following are examples of the description and the recommended action displayed by the Error Message Decoder tool for some of the error messages.

Error Message: `%PMAN-0-PROCESS_NOTIFICATION : The process lifecycle notification component failed because [chars]`	
Explanation	**Recommended Action**

The process lifecycle notification component failed, preventing proper detection of a process start and stop. This problem is likely the result of a software defect in the software subpackage.	Note the time of the message and investigate the kernel error message logs to learn more about the problem and see if it is correctable. If the problem cannot be corrected or the logs are not helpful, copy the error message exactly as it appears on the console along with the output of the **show tech-support** command and provide the gathered information to a Cisco technical support representative.

Error Message: `%PMAN-0-PROCFAILCRIT A critical process [chars] has failed (rc [dec])`

Explanation	Recommended Action
A process important to the functioning of the router has failed.	Note the time of the message and investigate the error message logs to learn more about the problem. If the problem persists, copy the message exactly as it appears on the console or in the system log. Research and attempt to resolve the issue using the tools and utilities provided at: http://www.cisco.com/tac. With some messages, these tools and utilities will supply clarifying information. Search for resolved software issues using the Bug Search Tool at: http://www.cisco.com/cisco/psn/bssprt/bss. If you still require assistance, open a case with the Technical Assistance Center at: http://tools.cisco.com/ServiceRequestTool/create/, or contact your Cisco technical support representative and provide the representative with the information you have gathered. Attach the following information to your case in nonzipped, plain-text (.txt) format: the output of the **show logging** and **show tech-support** commands and your pertinent troubleshooting logs.

Error Message: `%PMAN-3-PROCFAILOPT An optional process [chars] has failed (rc [dec])`

Explanation	Recommended Action

A process that does not affect the forwarding of traffic has failed.	Note the time of the message and investigate the kernel error message logs to learn more about the problem. Although traffic will still be forwarded after receiving this message, certain functions on the router may be disabled because of this message and the error should be investigated. If the logs are not helpful or indicate a problem you cannot correct, copy the message exactly as it appears on the console or in the system log. Research and attempt to resolve the issue using the tools and utilities provided at http://www.cisco.com/tac. With some messages, these tools and utilities will supply clarifying information. Search for resolved software issues using the Bug Search Tool at: http://www.cisco.com/cisco/psn/bssprt/bss. If you still require assistance, open a case with the Technical Assistance Center at: http://tools.cisco.com/ServiceRequestTool/create/, or contact your Cisco technical support representative and provide the representative with the information you have gathered. Attach the following information to your case in nonzipped, plain-text (.txt) format: the output of the **show logging** and **show tech-support** commands and your pertinent troubleshooting logs.

Error Message: `%PMAN-3-PROCFAIL The process [chars] has failed (rc [dec])`

Explanation	Recommended Action
The process has failed as the result of an error.	This message will appear with other messages related to the process. Check the other messages to determine the reason for the failures and see if corrective action can be taken. If the problem persists, copy the message exactly as it appears on the console or in the system log. Research and attempt to resolve the issue using the tools and utilities provided at: http://www.cisco.com/tac. With some messages, these tools and utilities will supply clarifying information. Search for resolved software issues using the Bug Search Tool at: http://www.cisco.com/cisco/psn/bssprt/bss. If you still require assistance, open a case with the Technical Assistance Center at: http://tools.cisco.com/ServiceRequestTool/create/, or contact your Cisco technical support representative and provide the representative with the information you have gathered. Attach the following information to your case in nonzipped, plain-text (.txt) format: the output of the **show logging** and **show tech-support** commands and your pertinent troubleshooting logs.

Error Message: `%PMAN-3-PROCFAIL_IGNORE [chars] process exits and failures are being ignored due to debug settings. Normal router functionality will be affected. Critical router functions like RP switchover, router reload, FRU resets, etc. may not function properly.`

Explanation	Recommended Action
A process failure is being ignored due to the user-configured debug settings.	If this behavior is desired and the debug settings are set according to a user's preference, no action is needed. If the appearance of this message is viewed as a problem, change the debug settings. The router is not expected to behave normally with this debug setting. Functionalities such as SSO switchover, router reloads, FRU resets, and so on will be affected. This setting should only be used in a debug scenario. It is not normal to run the router with this setting.

Error Message: `%PMAN-3-PROCHOLDDOWN The process [chars] has been helddown (rc [dec])`

Explanation	Recommended Action
The process was restarted too many times with repeated failures and has been placed in the hold-down state.	This message will appear with other messages related to the process. Check the other messages to determine the reason for the failures and see if corrective action can be taken. If the problem persists, copy the message exactly as it appears on the console or in the system log. Research and attempt to resolve the issue using the tools and utilities provided at: http://www.cisco.com/tac. With some messages, these tools and utilities will supply clarifying information. Search for resolved software issues using the Bug Search Tool at: http://www.cisco.com/cisco/psn/bssprt/bss. If you still require assistance, open a case with the Technical Assistance Center at: http://tools.cisco.com/ServiceRequestTool/create/, or contact your Cisco technical support representative and provide the representative with the information you have gathered. Attach the following information to your case in nonzipped, plain-text (.txt) format: the output of the **show logging** and **show tech-support** commands and your pertinent troubleshooting logs.

Error Message: `%PMAN-3-RELOAD_RP_SB_NOT_READY : Reloading: [chars]`

Explanation	Recommended Action
The route processor is being reloaded because there is no ready standby instance.	Ensure that the reload is not due to an error condition.

Error Message: `%PMAN-3-RELOAD_RP : Reloading: [chars]`

Explanation	Recommended Action

The RP is being reloaded.	Ensure that the reload is not due to an error condition. If it is due to an error condition, collect information requested by the other log messages.

Error Message: `%PMAN-3-RELOAD_SYSTEM : Reloading: [chars]`

Explanation	Recommended Action
The system is being reloaded.	Ensure that the reload is not due to an error condition. If it is due to an error condition, collect information requested by the other log messages.

Error Message: `%PMAN-3-PROC_BAD_EXECUTABLE : Bad executable or permission problem with process [chars]`

Explanation	Recommended Action
The executable file used for the process is bad or has permission problem.	Ensure that the named executable is replaced with the correct executable.

Error Message: `%PMAN-3-PROC_BAD_COMMAND:Non-existent executable or bad library used for process <process name>`

Explanation	Recommended Action
The executable file used for the process is missing, or a dependent library is bad.	Ensure that the named executable is present and the dependent libraries are good.

Error Message: `%PMAN-3-PROC_EMPTY_EXEC_FILE : Empty executable used for process [chars]`

Explanation	Recommended Action
The executable file used for the process is empty.	Ensure that the named executable is non-zero in size.

Error Message: `%PMAN-5-EXITACTION : Process manager is exiting: [chars]`

Explanation	Recommended Action
The process manager is exiting.	Ensure that the process manager is not exiting due to an error condition. If it is due to an error condition, collect information requested by the other log messages.

Error Message: `%PMAN-6-PROCSHUT : The process [chars] has shutdown`

Explanation	Recommended Action
The process has gracefully shut down.	No user action is necessary. This message is provided for informational purposes only.

Error Message: `%PMAN-6-PROCSTART : The process [chars] has started`

Explanation	Recommended Action

The process has launched and is operating properly.	No user action is necessary. This message is provided for informational purposes only.

Error Message: `%PMAN-6-PROCSTATELESS : The process [chars] is restarting stateless`

Explanation	Recommended Action
The process has requested a stateless restart.	No user action is necessary. This message is provided for informational purposes only.

CHAPTER 15

Trace Management

The following sections are included in this chapter:

Tracing Overview

Tracing is a function that logs internal events. Trace files containing trace messages are automatically created and saved to the tracelogs directory on the hard disk: file system on the router, which stores tracing files in bootflash.

The contents of trace files are useful for the following purposes:

- Troubleshooting—Helps to locate and solve an issue with a router. The trace files can be accessed in diagnostic mode even if other system issues are occurring simultaneously.
- Debugging—Helps to obtain a detailed view of system actions and operations.

How Tracing Works

Tracing logs the contents of internal events on a router. Trace files containing all the trace output pertaining to a module are periodically created and updated and stored in the tracelog directory. Trace files can be erased from this directory to recover space on the file system without impacting system performance. The files can be copied to other destinations using file transfer functions (such as FTP and TFTP) and opened using a plain text editor.

Note Tracing cannot be disabled on a router.

Use the following commands to view trace information and set tracing levels:

- **show platform software trace message**—Shows the most recent trace information for a specific module. This command can be used in privileged EXEC and diagnostic modes. When used in diagnostic mode, this command can gather trace log information during a Cisco IOS XE failure.
- **set platform software trace**—Sets a tracing level that determines the types of messages that are stored in the output. For more information on tracing levels, see Tracing Levels, on page 200.

Tracing Levels

Tracing levels determine how much information should be stored about a module in the trace buffer or file.

The following table shows all the tracing levels that are available and provides descriptions of what types of messages are displayed with each tracing level.

Table 22: Tracing Levels and Descriptions

Tracing Level	Level Number	Description
Emergency	0	The message is regarding an issue that makes the system unusable.
Alert	1	The message is regarding an action that must be taken immediately.
Critical	2	The message is regarding a critical condition. This is the default setting for every module on the router.
Error	3	The message is regarding a system error.
Warning	4	The message is regarding a system warning.
Notice	5	The message is regarding a significant issue, but the router is still working normally.
Informational	6	The message is useful for informational purposes only.
Debug	7	The message provides debug-level output.
Verbose	8	All possible tracing messages are sent.

Tracing Level	Level Number	Description
Noise	—	All possible trace messages pertaining to a module are logged. The noise level is always equal to the highest possible tracing level. Even if a future enhancement to tracing introduces a higher tracing level than verbose level, the noise level will become equal to the level of the newly introduced tracing level.

If a tracing level is set, messages are collected from both lower tracing levels and from its own level.

For example, setting the tracing level to 3 (error) means that the trace file will contain output messages for levels: 0 (emergencies), 1 (alerts), 2 (critical), and 3 (error).

If you set the trace level to 4 (warning), it results in output messages for levels: 0 (emergencies), 1 (alerts), 2 (critical), 3 (error), and 4 (warning).

The default tracing level for every module on the router is 5 (notice).

A tracing level is not set in a configuration mode, which results in tracing-level settings being returned to default values after the router reloads.

Caution Setting the tracing level of a module to debug level or higher can have a negative impact on the performance.

Caution Setting high tracing levels on a large number of modules can severely degrade performance. If a high tracing level is required in a specific context, it is almost always preferable to set the tracing level of a single module to a higher level rather than setting multiple modules to high levels.

Viewing a Tracing Level

By default, all the modules on a router are set to 5 (notice). This setting is maintained unless changed by a user.

To see the tracing level for a module on a router, enter the **show platform software trace level** command in privileged EXEC mode or diagnostic mode.

The following example shows how the **show platform software trace level** command is used to view the tracing levels of the forwarding manager processes on an active RP:

```
Router# show platform software trace level forwarding-manager rp active
Module Name                     Trace Level
-----------------------------------------------
acl                             Notice
binos                           Notice
binos/brand                     Notice
bipc                            Notice
```

```
bsignal                          Notice
btrace                           Notice
cce                              Notice
cdllib                           Notice
cef                              Notice
chasfs                           Notice
chasutil                         Notice
erspan                           Notice
ess                              Notice
ether-channel                    Notice
evlib                            Notice
evutil                           Notice
file_alloc                       Notice
fman_rp                          Notice
fpm                              Notice
fw                               Notice
icmp                             Notice
interfaces                       Notice
iosd                             Notice
ipc                              Notice
ipclog                           Notice
iphc                             Notice
IPsec                            Notice
mgmte-acl                        Notice
mlp                              Notice
mqipc                            Notice
nat                              Notice
nbar                             Notice
netflow                          Notice
om                               Notice
peer                             Notice
qos                              Notice
route-map                        Notice
sbc                              Notice
services                         Notice
sw_wdog                          Notice
tdl_acl_config_type              Notice
tdl_acl_db_type                  Notice
tdl_cdlcore_message              Notice
tdl_cef_config_common_type       Notice
tdl_cef_config_type              Notice
tdl_dpidb_config_type            Notice
tdl_fman_rp_comm_type            Notice
tdl_fman_rp_message              Notice
tdl_fw_config_type               Notice
tdl_hapi_tdl_type                Notice
tdl_icmp_type                    Notice
tdl_ip_options_type              Notice
tdl_ipc_ack_type                 Notice
tdl_IPsec_db_type                Notice
tdl_mcp_comm_type                Notice
tdl_mlp_config_type              Notice
tdl_mlp_db_type                  Notice
tdl_om_type                      Notice
tdl_ui_message                   Notice
tdl_ui_type                      Notice
tdl_urpf_config_type             Notice
tdllib                           Notice
trans_avl                        Notice
uihandler                        Notice
uipeer                           Notice
uistatus                         Notice
urpf                             Notice
vista                            Notice
```

```
wccp                                Notice
```

Setting a Tracing Level

To set a tracing level for a module on a router, or for all the modules within a process on a router, enter the **set platform software trace** command in the privileged EXEC mode or diagnostic mode.

The following example shows the tracing level for the ACL module in the Forwarding Manager of the ESP processor in slot 0 set to `info`:

```
set platform software trace forwarding-manager F0 acl info
```

Viewing the Content of the Trace Buffer

To view the trace messages in the trace buffer or file, enter the **show platform software trace message** command in privileged EXEC or diagnostic mode. In the following example, the trace messages for the Host Manager process in Route Processor slot 0 are viewed using the **show platform software trace message command**:

```
Router# show platform software trace message host-manager R0
08/23 12:09:14.408 [uipeer]: (info): Looking for a ui_req msg
08/23 12:09:14.408 [uipeer]: (info): Start of request handling for con 0x100a61c8
08/23 12:09:14.399 [uipeer]: (info): Accepted connection for 14 as 0x100a61c8
08/23 12:09:14.399 [uipeer]: (info): Received new connection 0x100a61c8 on descriptor 14
08/23 12:09:14.398 [uipeer]: (info): Accepting command connection on listen fd 7
08/23 11:53:57.440 [uipeer]: (info): Going to send a status update to the shell manager in
 slot 0
08/23 11:53:47.417 [uipeer]: (info): Going to send a status update to the shell manager in
 slot 0
```

CHAPTER 16

Packet Trace

First Published: August 03, 2016

The Packet-Trace feature provides a detailed understanding of how data packets are processed by the Cisco IOS XE platform, and thus helps customers to diagnose issues and troubleshoot them more efficiently. This module provides information about how to use the Packet-Trace feature.

Information About Packet Trace

The Packet-Trace feature provides three levels of inspection for packets: accounting, summary, and path data. Each level provides a detailed view of packet processing at the cost of some packet processing capability. However, Packet Trace limits inspection to packets that match the debug platform condition statements, and is a viable option even under heavy-traffic situations in customer environments.

The following table explains the three levels of inspection provided by packet trace.

Table 23: Packet-Trace Level

Packet-Trace Level	Description
Accounting	Packet-Trace accounting provides a count of packets that enter and leave the network processor. Packet-Trace accounting is a lightweight performance activity, and runs continuously until it is disabled.
Summary	At the summary level of packet trace, data is collected for a finite number of packets. Packet-Trace summary tracks the input and output interfaces, the final packet state, and punt, drop, or inject packets, if any. Collecting summary data adds to additional performance compared to normal packet processing, and can help to isolate a troublesome interface.

Packet-Trace Level	Description
Path data	The packet-trace path data level provides the greatest level of detail in packet trace. Data is collected for a finite number of packets. Packet-Trace path data captures data, including a conditional debugging ID that is useful to correlate with feature debugs, a timestamp, and also feature-specific path-trace data. Path data also has two optional capabilities: packet copy and Feature Invocation Array (FIA) trace. The packet-copy option enables you to copy input and output packets at various layers of the packet (layer 2, layer 3 or layer 4). The FIA- trace option tracks every feature entry invoked during packet processing and helps you to know what is happening during packet processing. **Note** Collecting path data consumes more packet-processing resources, and the optional capabilities incrementally affect packet performance. Therefore, path-data level should be used in limited capacity or in situations where packet performance change is acceptable.

Usage Guidelines for Configuring Packet Trace

Consider the following best practices while configuring the Packet-Trace feature:

- Use of ingress conditions when using the Packet-Trace feature is recommended for a more comprehensive view of packets.
- Packet-trace configuration requires data-plane memory. On systems where data-plane memory is constrained, carefully consider how you will select the packet-trace values. A close approximation of the amount of memory consumed by packet trace is provided by the following equation:

memory required = (statistics overhead) + number of packets * (summary size + data size + packet copy size).

When the Packet-Trace feature is enabled, a small, fixed amount of memory is allocated for statistics. Similarly, when per-packet data is captured, a small, fixed amount of memory is required for each packet for summary data. However, as shown by the equation, you can significantly influence the amount of memory consumed by the number of packets you select to trace, and whether you collect path data and copies of packets.

Configuring Packet Trace

Perform the following steps to configure the Packet-Trace feature.

Note The amount of memory consumed by the Packet-Trace feature is affected by the packet-trace configuration. You should carefully select the size of per-packet path data and copy buffers and the number of packets to be traced in order to avoid interrupting normal services. You can check the current data-plane DRAM memory consumption by using the **show platform hardware qfp active infrastructure exmem statistics** command.

SUMMARY STEPS

1. **enable**
2. **debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [circular] [data-size** *data-size***]**
3. **debug platform packet-trace {punt |inject|copy|drop|packet|statistics}**
4. **debug platform condition [ipv4 | ipv6] [interface** *interface***][access-list** *access-list -name* | *ipv4-address / subnet-mask* | *ipv6-address / subnet-mask*] **[ingress | egress |both]**
5. **debug platform condition start**
6. **debug platform condition stop**
7. **show platform packet-trace {configuration | statistics | summary | packet {all |** *pkt-num***}}**
8. **clear platform condition all**
9. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [circular] [data-size** *data-size***]** **Example:** `Router# debug platform packet-trace packets 2048 summary-only`	Collects summary data for a specified number of packets. Captures feature path data by default, and optionally performs FIA trace. *pkt-num*—Specifies the maximum number of packets maintained at a given time. **fia-trace**—Provides detailed level of data capture, including summary data, feature-specific data. Also displays each feature entry visited during packet processing. **summary-only**—Enables the capture of summary data with minimal details. **circular**—Saves the data of the most recently traced packets. *data-size*—Specifies the size of data buffers for storing feature and FIA trace data for each packet in bytes. When very heavy packet processing is performed on packets, users can increase the size of the data buffers if necessary. The default value is 2048.
Step 3	**debug platform packet-trace {punt |inject|copy|drop|packet|statistics}** **Example:** `Router# debug platform packet-trace punt`	Enables tracing of punted packets from data to control plane.

	Command or Action	Purpose
Step 4	**debug platform condition [ipv4 | ipv6] [interface** *interface***][access-list** *access-list -name* | *ipv4-address* / *subnet-mask* | *ipv6-address* / *subnet-mask***] [ingress | egress |both]** **Example:** `Router# debug platform condition interface g0/0/0 ingress`	Specifies the matching criteria for tracing packets. Provides the ability to filter by protocol, IP address and subnet mask, access control list (ACL), interface, and direction.
Step 5	**debug platform condition start** **Example:** `Router# debug platform condition start`	Enables the specified matching criteria and starts packet tracing.
Step 6	**debug platform condition stop** **Example:** `Router# debug platform condition start`	Deactivates the condition and stops packet tracing.
Step 7	**show platform packet-trace {configuration | statistics | summary | packet {all |** *pkt-num***}}** **Example:** `Router# show platform packet-trace 14`	Displays packet-trace data according to the specified option. See {start cross reference} Table 21-1 {end cross reference} for detailed information about the **show** command options.
Step 8	**clear platform condition all** **Example:** `Router(config)# clear platform condition all`	Removes the configurations provided by the **debug platform condition** and **debug platform packet-trace** commands.
Step 9	**exit** **Example:** `Router# exit`	Exits the privileged EXEC mode.

Configuring Packet Tracer with UDF Offset

Perform the following steps to configure the Packet-Trace UDF with offset:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **udf** *udf name* **header** {**inner** | **outer**} {**13**|**14**} **offset** *offset-in-bytes* **length** *length-in-bytes*
4. **udf** *udf name* {**header** | **packet-start**} *offset-base offset length*
5. **ip access-list extended** {*acl-name* |*acl-num*}

6. **ip access-list extended { deny | permit } udf udf-name value mask**
7. **debug platform condition [ipv4 | ipv6] [interface** *interface***] [access-list** *access-list -name* | *ipv4-address* / *subnet-mask* | *ipv6-address* / *subnet-mask***] [ingress | egress |both]**
8. **debug platform condition start**
9. **debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [circular] [data-size** *data-size***]**
10. **debug platform packet-trace {punt | inject|copy | drop |packet | statistics}**
11. **debug platform condition stop**
12. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**udf** *udf name* **header** {**inner** | **outer**} {**13|14**} **offset** *offset-in-bytes* **length** *length-in-bytes*
	Example:
	`Router(config)# udf TEST_UDF_NAME_1 header inner l3 64 1`
	`Router(config)# udf TEST_UDF_NAME_2 header inner l4 77 2`
	`Router(config)# udf TEST_UDF_NAME_3 header outer l3 65 1`
	`Router(config)# udf TEST_UDF_NAME_4 header outer l4 67 1`
		Configures individual UDF definitions. You can specify the name of the UDF, the networking header from which offset, and the length of data to be extracted.
		The **inner** or **outer** keywords indicate the start of the offset from the unencapsulated Layer 3 or Layer 4 headers, or if there is an encapsulated packet, they indicate the start of offset from the inner L3/L4.
		The **length** keyword specifies, in bytes, the length from the offset. The range is from 1 to 2.
		.

Step 4	**udf** *udf name* {**header** | **packet-start**} *offset-base offset length* **Example:** `Router(config)# udf TEST_UDF_NAME_5 packet-start 120 1`	• header—Specifies the offset base configuration. • packet-start—Specifies the offset base from packet-start. packet-start" can vary depending on if packet-trace is for an inbound packet or outbound packet. If the packet-trace is for an inbound packet then the packet-start will be layer2. For outbound, he packet-start will be layer3. • offset—Specifies the number of bytes offset from the offset base. To match the first byte from the offset

	Command or Action	Purpose
		base (Layer 3/Layer 4 header), configure the offset as 0. • length—Specifies the number of bytes from the offset. Only 1 or 2 bytes are supported. To match additional bytes, you must define multiple UDFs.
Step 5	**ip access-list extended** {*acl-name* |*acl-num*} **Example:** `Router(config)# ip access-list extended acl2`	Enables extended ACL configuration mode. The CLI enters the extended ACL configuration mode in which all subsequent commands apply to the current extended access list. Extended ACLs control traffic by the comparison of the source and destination addresses of the IP packets to the addresses configured in the ACL.
Step 6	**ip access-list extended { deny | permit } udf udf-name value mask** **Example:** `Router(config-acl)# permit ip any any udf TEST_UDF_NAME_5 0xD3 0xFF`	Configures the ACL to match on UDFs along with the current access control entries (ACEs) . The bytes defined in ACL is 0xD3. Masks are used with IP addresses in IP ACLs to specify what should be permitted and denied.
Step 7	**debug platform condition [ipv4 | ipv6] [interface** *interface*] **[access-list** *access-list -name* | *ipv4-address* / *subnet-mask* | *ipv6-address* / *subnet-mask*] **[ingress | egress |both]** **Example:** `Router# debug platform condition interface gi0/0/0 ipv4 access-list acl2 both`	Specifies the matching criteria for tracing packets. Provides the ability to filter by protocol, IP address and subnet mask, access control list (ACL), interface, and direction.
Step 8	**debug platform condition start** **Example:** `Router# debug platform condition start`	Enables the specified matching criteria and starts packet tracing.
Step 9	**debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [circular] [data-size** *data-size*] **Example:** `Router# debug platform packet-trace packet 1024 fia-trace data-size 2048`	Collects summary data for a specified number of packets. Captures feature path data by default, and optionally performs FIA trace. *pkt-num*—Specifies the maximum number of packets maintained at a given time. **fia-trace**—Provides detailed level of data capture, including summary data, feature-specific data. Also displays each feature entry visited during packet processing. **summary-only**—Enables the capture of summary data with minimal details. **circular**—Saves the data of the most recently traced packets.

	Command or Action	Purpose
		data-size—Specifies the size of data buffers for storing feature and FIA trace data for each packet in bytes. When very heavy packet processing is performed on packets, users can increase the size of the data buffers if necessary. The default value is 2048.
Step 10	**debug platform packet-trace {punt | inject|copy | drop |packet | statistics}** **Example:** `Router# debug platform packet-trace punt`	Enables tracing of punted packets from data to control plane.
Step 11	**debug platform condition stop** **Example:** `Router# debug platform condition start`	Deactivates the condition and stops packet tracing.
Step 12	**exit** **Example:** `Router# exit`	Exits the privileged EXEC mode.

Displaying Packet-Trace Information

Use these **show** commands to display packet-trace information.

Table 24: show Commands

Command	Description
show platform packet-trace configuration	Displays packet trace configuration, including any defaults.
show platform packet-trace statistics	Displays accounting data for all the traced packets.
show platform packet-trace summary	Displays summary data for the number of packets specified.
show platform packet-trace {all | *pkt-num***} [decode]**	Displays the path data for all the packets or the packet specified. The **decode** option attempts to decode the binary packet into a more human- readable form.

Removing Packet-Trace Data

Use these commands to clear packet-trace data.

Table 25: clear Commands

Command	Description
clear platform packet-trace statistics	Clears the collected packet-trace data and statistics.
clear platform packet-trace configuration	Clears the packet-trace configuration and the statistics.

Configuration Examples for Packet Trace

This section provides the following configuration examples:

Example: Configuring Packet Trace

This example describes how to configure packet trace and display the results. In this example, incoming packets to Gigabit Ethernet interface 0/0/1 are traced, and FIA-trace data is captured for the first 128 packets. Also, the input packets are copied. The **show platform packet-trace packet 0** command displays the summary data and each feature entry visited during packet processing for packet 0.

```
Router>
enable
Router# debug platform packet-trace packet 128 fia-trace
Router# debug platform packet-trace punt
Router# debug platform condition interface g0/0/1 ingress
Router# debug platform condition start
Router#! ping to UUT
Router# debug platform condition stop
Router# show platform packet-trace packet 0
Packet: 0           CBUG ID: 9
Summary
  Input     : GigabitEthernet0/0/1
  Output    : GigabitEthernet0/0/0
  State     : FWD
  Timestamp
    Start   : 1819281992118 ns (05/17/2014 06:42:01.207240 UTC)
    Stop    : 1819282095121 ns (05/17/2014 06:42:01.207343 UTC)
Path Trace
  Feature: IPV4
    Source      : 192.0.2.1
    Destination : 192.0.2.2
    Protocol    : 1 (ICMP)
  Feature: FIA_TRACE
    Entry     : 0x8059dbe8 - DEBUG_COND_INPUT_PKT
    Timestamp : 3685243309297
  Feature: FIA_TRACE
    Entry     : 0x82011a00 - IPV4_INPUT_DST_LOOKUP_CONSUME
    Timestamp : 3685243311450
  Feature: FIA_TRACE
    Entry     : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
    Timestamp : 3685243312427
  Feature: FIA_TRACE
    Entry     : 0x82004b68 - IPV4_OUTPUT_LOOKUP_PROCESS
    Timestamp : 3685243313230
  Feature: FIA_TRACE
    Entry     : 0x8034f210 - IPV4_INPUT_IPOPTIONS_PROCESS
    Timestamp : 3685243315033
```

```
  Feature: FIA_TRACE
    Entry     : 0x82013200 - IPV4_OUTPUT_GOTO_OUTPUT_FEATURE
    Timestamp : 3685243315787
  Feature: FIA_TRACE
    Entry     : 0x80321450 - IPV4_VFR_REFRAG
    Timestamp : 3685243316980
  Feature: FIA_TRACE
    Entry     : 0x82014700 - IPV6_INPUT_L2_REWRITE
    Timestamp : 3685243317713
  Feature: FIA_TRACE
    Entry     : 0x82000080 - IPV4_OUTPUT_FRAG
    Timestamp : 3685243319223
  Feature: FIA_TRACE
    Entry     : 0x8200e500 - IPV4_OUTPUT_DROP_POLICY
    Timestamp : 3685243319950
  Feature: FIA_TRACE
    Entry     : 0x8059aff4 - PACTRAC_OUTPUT_STATS
    Timestamp : 3685243323603
  Feature: FIA_TRACE
    Entry     : 0x82016100 - MARMOT_SPA_D_TRANSMIT_PKT
    Timestamp : 3685243326183

Router# clear platform condition all
Router# exit
```

Linux Forwarding Transport Service (LFTS) is a transport mechanism to forward packets punted from the CPP into applications other than IOSd. This example displays the LFTS-based intercepted packet destined for binos application.

```
Router# show platform packet-trace packet 10
Packet: 10     CBUG ID: 52
Summary
  Input  : GigabitEthernet0/0/0
  Output : internal0/0/rp:1
  State  : PUNT 55 (For-us control)
  Timestamp
    Start : 597718358383 ns (06/06/2016 09:00:13.643341 UTC)
    Stop : 597718409650 ns (06/06/2016 09:00:13.643392 UTC)
Path Trace
  Feature: IPV4
    Input  : GigabitEthernet0/0/0
    Output : <unknown>
    Source : 10.64.68.2
    Destination : 10.0.0.102
    Protocol : 17 (UDP)
      SrcPort : 1985
      DstPort : 1985
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : <unknown>
    Entry  : 0x8a0177bc - DEBUG_COND_INPUT_PKT
    Lapsed time : 426 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : <unknown>
    Entry  : 0x8a017788 - IPV4_INPUT_DST_LOOKUP_CONSUME
    Lapsed time : 386 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : <unknown>
    Entry  : 0x8a01778c - IPV4_INPUT_FOR_US_MARTIAN
    Lapsed time : 13653 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
```

```
    Output : internal0/0/rp:1
    Entry  : 0x8a017730 - IPV4_INPUT_LOOKUP_PROCESS_EXT
    Lapsed time : 2360 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : internal0/0/rp:1
    Entry  : 0x8a017be0 - IPV4_INPUT_IPOPTIONS_PROCESS_EXT
    Lapsed time : 66 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : internal0/0/rp:1
    Entry  : 0x8a017bfc - IPV4_INPUT_GOTO_OUTPUT_FEATURE_EXT
    Lapsed time : 680 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : internal0/0/rp:1
    Entry  : 0x8a017d60 - IPV4_INTERNAL_ARL_SANITY_EXT
    Lapsed time : 320 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : internal0/0/rp:1
    Entry  : 0x8a017a40 - IPV4_VFR_REFRAG_EXT
    Lapsed time : 106 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : internal0/0/rp:1
    Entry  : 0x8a017d2c - IPV4_OUTPUT_DROP_POLICY_EXT
    Lapsed time : 1173 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : internal0/0/rp:1
    Entry  : 0x8a017940 - INTERNAL_TRANSMIT_PKT_EXT
    Lapsed time : 20173 ns
  LFTS Path Flow: Packet: 10    CBUG ID: 52
    Feature: LFTS
    Pkt Direction: IN
    Punt Cause  : 55
         subCause : 0
```

Example: Using Packet Trace

This example provides a scenario in which packet trace is used to troubleshoot packet drops for a NAT configuration on a Cisco device. This example shows how you can effectively utilize the level of detail provided by the Packet-Trace feature to gather information about an issue, isolate the issue, and then find a solution.

In this scenario, you can detect that there are issues, but are not sure where to start troubleshooting. You should, therefore, consider accessing the Packet-Trace summary for a number of incoming packets.

```
Router# debug platform condition ingress
Router# debug platform packet-trace packet 2048 summary-only
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary
Pkt   Input             Output            State  Reason
0     Gi0/0/0           Gi0/0/0           DROP   402 (NoStatsUpdate)
1     internal0/0/rp:0  internal0/0/rp:0  PUNT   21  (RP<->QFP keepalive)
2     internal0/0/recycle:0  Gi0/0/0      FWD
```

The output shows that packets are dropped due to NAT configuration on Gigabit Ethernet interface 0/0/0, which enables you to understand that an issue is occurring on a specific interface. Using this information, you can limit which packets to trace, reduce the number of packets for data capture, and increase the level of inspection.

```
Router# debug platform packet-trace packet 256
Router# debug platform packet-trace punt
Router# debug platform condition interface Gi0/0/0
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary
Router# show platform packet-trace 15
Packet: 15          CBUG ID: 238
Summary
  Input     : GigabitEthernet0/0/0
  Output    : internal0/0/rp:1
  State     : PUNT 55  (For-us control)
  Timestamp
    Start   : 1166288346725 ns (06/06/2016 09:09:42.202734 UTC)
    Stop    : 1166288383210 ns (06/06/2016 09:09:42.202770 UTC)
Path Trace
  Feature: IPV4
    Input       : GigabitEthernet0/0/0
    Output      : <unknown>
    Source      : 10.64.68.3
    Destination : 10.0.0.102
    Protocol    : 17 (UDP)
      SrcPort   : 1985
      DstPort   : 1985
IOSd Path Flow: Packet: 15    CBUG ID: 238
  Feature: INFRA
    Pkt Direction: IN
    Packet Rcvd From CPP
  Feature: IP
    Pkt Direction: IN
    Source      : 10.64.68.122
    Destination : 10.64.68.255
  Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.64.68.122
    Destination : 10.64.68.255
    Interface   : GigabitEthernet0/0/0
  Feature: UDP
    Pkt Direction: IN
    src         : 10.64.68.122(1053)
    dst         : 10.64.68.255(1947)
    length      : 48

Router#show platform packet-trace packet 10
Packet: 10          CBUG ID: 10
Summary
  Input     : GigabitEthernet0/0/0
  Output    : internal0/0/rp:0
  State     : PUNT 55  (For-us control)
  Timestamp
    Start   : 274777907351 ns (01/10/2020 10:56:47.918494 UTC)
    Stop    : 274777922664 ns (01/10/2020 10:56:47.918509 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet0/0/0
    Output      : <unknown>
    Source      : 10.78.106.2
    Destination : 10.0.0.102
    Protocol    : 17 (UDP)
```

```
      SrcPort   : 1985
      DstPort   : 1985

IOSd Path Flow: Packet: 10    CBUG ID: 10
  Feature: INFRA
    Pkt Direction: IN
Packet Rcvd From DATAPLANE
 Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.78.106.2
    Destination : 10.0.0.102
    Interface   : GigabitEthernet0/0/0

  Feature: UDP
    Pkt Direction: IN DROP
    Pkt : DROPPED
    UDP: Discarding silently
    src        : 881 10.78.106.2(1985)
    dst         : 10.0.0.102(1985)
    length      : 60

Router#show platform packet-trace packet  12
Packet: 12          CBUG ID: 767
Summary
  Input     : GigabitEthernet3
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 16120990774814 ns (01/20/2020 12:38:02.816435 UTC)
    Stop    : 16120990801840 ns (01/20/2020 12:38:02.816462 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet3
    Output      : <unknown>
    Source      : 10.1.1.1
    Destination : 10.1.1.2
    Protocol    : 6 (TCP)
      SrcPort   : 46593
      DstPort   : 23
IOSd Path Flow: Packet: 12    CBUG ID: 767
  Feature: INFRA
    Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.1.1.1
    Destination : 10.1.1.2
    Interface   : GigabitEthernet3

  Feature: IP
    Pkt Direction: IN
    FORWARDEDTo transport layer
    Source        : 10.1.1.1
    Destination   : 10.1.1.2
    Interface     : GigabitEthernet3

  Feature: TCP
    Pkt Direction: IN
    tcp0: I NoTCB 10.1.1.1:46593 10.1.1.2:23 seq 1925377975 OPTS 4 SYN  WIN 4128

Router# show platform packet-trace summary
Pkt   Input                     Output                    State  Reason
```

```
0      INJ.2                    Gi1                       FWD
1      Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
2      INJ.2                    Gi1                       FWD
3      Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
4      INJ.2                    Gi1                       FWD
5      INJ.2                    Gi1                       FWD
6      Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
7      Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
8      Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
9      Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
10     INJ.2                    Gi1                       FWD
11     INJ.2                    Gi1                       FWD
12     INJ.2                    Gi1                       FWD
13     Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
14     Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
15     Gi1                      internal0/0/rp:0          PUNT   11  (For-us data)
16     INJ.2                    Gi1                       FWD
```

The following example displays the packet trace data statistics.

```
Router#show platform packet-trace statistics
Packets Summary
  Matched  3
  Traced   3
Packets Received
  Ingress  0
  Inject   0
Packets Processed
  Forward  0
  Punt     3
    Count       Code  Cause
    3           56    RP injected for-us control
  Drop     0
  Consume  0

          PKT_DIR_IN
             Dropped       Consumed       Forwarded
INFRA           0             0              0
TCP             0             0              0
UDP             0             0              0
IP              0             0              0
IPV6            0             0              0
ARP             0             0              0

         PKT_DIR_OUT
             Dropped       Consumed       Forwarded
INFRA           0             0              0
TCP             0             0              0
UDP             0             0              0
IP              0             0              0
IPV6            0             0              0
ARP             0             0              0
```

The following example displays packets that are injected and punted to the forwarding processor from the control plane.

```
Router#debug platform condition ipv4 10.118.74.53/32 both
 Router#Router#debug platform condition start
Router#debug platform packet-trace packet 200
Packet count rounded up from 200 to 256

Router#show platform packet-tracer packet 0
show plat pack pa 0
Packet: 0           CBUG ID: 674
Summary
```

```
  Input     : GigabitEthernet1
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 17756544435656 ns (06/29/2020 18:19:17.326313 UTC)
    Stop    : 17756544469451 ns (06/29/2020 18:19:17.326346 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet1
    Output      : <unknown>
    Source      : 10.118.74.53
    Destination : 172.18.124.38
    Protocol    : 17 (UDP)
      SrcPort   : 2640
      DstPort   : 500

IOSd Path Flow: Packet: 0    CBUG ID: 674
  Feature: INFRA
  Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
  Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.118.74.53
    Destination : 172.18.124.38
    Interface   : GigabitEthernet1

  Feature: IP
  Pkt Direction: IN
  FORWARDED To transport layer
    Source        : 10.118.74.53
    Destination   : 172.18.124.38
    Interface     : GigabitEthernet1

  Feature: UDP
  Pkt Direction: IN
  DROPPED
 UDP: Checksum error: dropping
 Source      : 10.118.74.53(2640)
 Destination : 172.18.124.38(500)

Router#show platform packet-tracer packet 2
Packet: 2           CBUG ID: 2

IOSd Path Flow:
  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 172.18.124.38:22 172.18.124.55:52774 seq 3052140910
OPTS 4 ACK 2346709419 SYN  WIN 4128

  Feature: TCP
  Pkt Direction: OUT
  FORWARDED
 TCP: Connection is in SYNRCVD state
 ACK         : 2346709419
 SEQ         : 3052140910
 Source      : 172.18.124.38(22)
 Destination : 172.18.124.55(52774)


  Feature: IP
  Pkt Direction: OUTRoute out the generated packet.srcaddr: 172.18.124.38, dstaddr:
172.18.124.55
```

```
  Feature: IP
  Pkt Direction: OUTInject and forward successful srcaddr: 172.18.124.38, dstaddr:
172.18.124.55

  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 172.18.124.38:22 172.18.124.55:52774 seq 3052140910
OPTS 4 ACK 2346709419 SYN  WIN 4128
Summary
  Input     : INJ.2
  Output    : GigabitEthernet1
  State     : FWD
  Timestamp
    Start   : 490928006866 ns (06/29/2020 13:31:30.807879 UTC)
    Stop    : 490928038567 ns (06/29/2020 13:31:30.807911 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : internal0/0/rp:0
    Output      : <unknown>
    Source      : 172.18.124.38
    Destination : 172.18.124.55
    Protocol    : 6 (TCP)
      SrcPort   : 22
      DstPort   : 52774
  Feature: IPSec
    Result    : IPSEC_RESULT_DENY
    Action    : SEND_CLEAR
    SA Handle : 0
    Peer Addr : 10.124.18.172
    Local Addr: 10.124.18.172


Router#
```

Example: Using Packet Trace

This example provides a scenario in which packet trace is used to troubleshoot packet drops for a NAT configuration on a Cisco ASR 1006 Router. This example shows how you can effectively utilize the level of detail provided by the Packet-Trace feature to gather information about an issue, isolate the issue, and then find a solution.

In this scenario, you can detect that there are issues, but are not sure where to start troubleshooting. You should, therefore, consider accessing the Packet-Trace summary for a number of incoming packets.

```
Router# debug platform condition ingress
Router# debug platform packet-trace packet 2048 summary-only
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary
Pkt   Input             Output            State  Reason
0     Gi0/0/0           Gi0/0/0           DROP   402 (NoStatsUpdate)
1     internal0/0/rp:0  internal0/0/rp:0  PUNT   21  (RP<->QFP keepalive)
2     internal0/0/recycle:0  Gi0/0/0      FWD
```

The output shows that packets are dropped due to NAT configuration on Gigabit Ethernet interface 0/0/0, which enables you to understand that an issue is occurring on a specific interface. Using this information, you can limit which packets to trace, reduce the number of packets for data capture, and increase the level of inspection.

```
Router# debug platform packet-trace packet 256
Router# debug platform packet-trace punt
```

```
Router# debug platform condition interface Gi0/0/0
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary
Router# show platform packet-trace 15
Packet: 15          CBUG ID: 238
Summary
  Input     : GigabitEthernet0/0/0
  Output    : internal0/0/rp:1
  State     : PUNT 55  (For-us control)
  Timestamp
    Start   : 1166288346725 ns (06/06/2016 09:09:42.202734 UTC)
    Stop    : 1166288383210 ns (06/06/2016 09:09:42.202770 UTC)
Path Trace
  Feature: IPV4
    Input       : GigabitEthernet0/0/0
    Output      : <unknown>
    Source      : 10.64.68.3
    Destination : 224.0.0.102
    Protocol    : 17 (UDP)
      SrcPort   : 1985
      DstPort   : 1985
IOSd Path Flow: Packet: 15    CBUG ID: 238
  Feature: INFRA
    Pkt Direction: IN
    Packet Rcvd From CPP
  Feature: IP
    Pkt Direction: IN
    Source      : 10.64.68.122
    Destination : 10.64.68.255
  Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.64.68.122
    Destination : 10.64.68.255
    Interface   : GigabitEthernet0/0/0
  Feature: UDP
    Pkt Direction: IN
    src         : 10.64.68.122(1053)
    dst         : 10.64.68.255(1947)
    length      : 48

Router#show platform packet-trace packet 10
Packet: 10          CBUG ID: 10
Summary
  Input     : GigabitEthernet0/0/0
  Output    : internal0/0/rp:0
  State     : PUNT 55  (For-us control)
  Timestamp
    Start   : 274777907351 ns (01/10/2020 10:56:47.918494 UTC)
    Stop    : 274777922664 ns (01/10/2020 10:56:47.918509 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet0/0/0
    Output      : <unknown>
    Source      : 10.78.106.2
    Destination : 224.0.0.102
    Protocol    : 17 (UDP)
      SrcPort   : 1985
      DstPort   : 1985

IOSd Path Flow: Packet: 10    CBUG ID: 10
  Feature: INFRA
    Pkt Direction: IN
```

```
Packet Rcvd From DATAPLANE
 Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.78.106.2
    Destination : 224.0.0.102
    Interface   : GigabitEthernet0/0/0

  Feature: UDP
    Pkt Direction: IN DROP
    Pkt : DROPPED
    UDP: Discarding silently
    src        : 881 10.78.106.2(1985)
    dst         : 224.0.0.102(1985)
    length      : 60

Router#show platform packet-trace packet  12
Packet: 12          CBUG ID: 767
Summary
  Input     : GigabitEthernet3
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 16120990774814 ns (01/20/2020 12:38:02.816435 UTC)
    Stop    : 16120990801840 ns (01/20/2020 12:38:02.816462 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet3
    Output      : <unknown>
    Source      : 12.1.1.1
    Destination : 12.1.1.2
    Protocol    : 6 (TCP)
      SrcPort   : 46593
      DstPort   : 23
IOSd Path Flow: Packet: 12    CBUG ID: 767
  Feature: INFRA
    Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 12.1.1.1
    Destination : 12.1.1.2
    Interface   : GigabitEthernet3

  Feature: IP
    Pkt Direction: IN
    FORWARDEDTo transport layer
    Source        : 12.1.1.1
    Destination   : 12.1.1.2
    Interface     : GigabitEthernet3

  Feature: TCP
    Pkt Direction: IN
    tcp0: I NoTCB 12.1.1.1:46593 12.1.1.2:23 seq 1925377975 OPTS 4 SYN  WIN 4128

Router# show platform packet-trace summary
Pkt   Input                     Output                    State  Reason
0     INJ.2                     Gi1                       FWD
1     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
2     INJ.2                     Gi1                       FWD
3     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
4     INJ.2                     Gi1                       FWD
5     INJ.2                     Gi1                       FWD
```

```
6     Gi1                        internal0/0/rp:0          PUNT   11  (For-us data)
7     Gi1                        internal0/0/rp:0          PUNT   11  (For-us data)
8     Gi1                        internal0/0/rp:0          PUNT   11  (For-us data)
9     Gi1                        internal0/0/rp:0          PUNT   11  (For-us data)
10    INJ.2                      Gi1                       FWD
11    INJ.2                      Gi1                       FWD
12    INJ.2                      Gi1                       FWD
13    Gi1                        internal0/0/rp:0          PUNT   11  (For-us data)
14    Gi1                        internal0/0/rp:0          PUNT   11  (For-us data)
15    Gi1                        internal0/0/rp:0          PUNT   11  (For-us data)
16    INJ.2                      Gi1                       FWD
```

The following example displays the packet trace data statistics.

```
Router#show platform packet-trace statistics
Packets Summary
  Matched  3
  Traced   3
Packets Received
  Ingress  0
  Inject   0
Packets Processed
  Forward  0
  Punt     3
    Count       Code  Cause
    3           56    RP injected for-us control
  Drop     0
  Consume  0

          PKT_DIR_IN
             Dropped       Consumed       Forwarded
INFRA           0             0              0
TCP             0             0              0
UDP             0             0              0
IP              0             0              0
IPV6            0             0              0
ARP             0             0              0

         PKT_DIR_OUT
             Dropped       Consumed       Forwarded
INFRA           0             0              0
TCP             0             0              0
UDP             0             0              0
IP              0             0              0
IPV6            0             0              0
ARP             0             0              0
```

The following example displays packets that are injected and punted to the forwarding processor from the control plane.

```
Router#debug platform condition ipv4 10.118.74.53/32 both
 Router#Router#debug platform condition start
Router#debug platform packet-trace packet 200
Packet count rounded up from 200 to 256

Router#show platform packet-tracer packet 0
show plat pack pa 0
Packet: 0           CBUG ID: 674
Summary
  Input     : GigabitEthernet1
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 17756544435656 ns (06/29/2020 18:19:17.326313 UTC)
    Stop    : 17756544469451 ns (06/29/2020 18:19:17.326346 UTC)
```

```
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet1
    Output      : <unknown>
    Source      : 10.118.74.53
    Destination : 198.51.100.38
    Protocol    : 17 (UDP)
      SrcPort   : 2640
      DstPort   : 500

IOSd Path Flow: Packet: 0    CBUG ID: 674
  Feature: INFRA
  Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
  Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.118.74.53
    Destination : 198.51.100.38
    Interface   : GigabitEthernet1

  Feature: IP
  Pkt Direction: IN
  FORWARDED To transport layer
    Source        : 10.118.74.53
    Destination   : 198.51.100.38
    Interface     : GigabitEthernet1

  Feature: UDP
  Pkt Direction: IN
  DROPPED
 UDP: Checksum error: dropping
 Source      : 10.118.74.53(2640)
 Destination : 198.51.100.38(500)

Router#show platform packet-tracer packet 2
Packet: 2           CBUG ID: 2

IOSd Path Flow:
  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 198.51.100.38:22 198.51.100.55:52774 seq 3052140910
OPTS 4 ACK 2346709419 SYN  WIN 4128

  Feature: TCP
  Pkt Direction: OUT
  FORWARDED
 TCP: Connection is in SYNRCVD state
 ACK         : 2346709419
 SEQ         : 3052140910
 Source      : 198.51.100.38(22)
 Destination : 198.51.100.55(52774)


  Feature: IP
  Pkt Direction: OUTRoute out the generated packet.srcaddr: 198.51.100.38, dstaddr:
198.51.100.55

  Feature: IP
  Pkt Direction: OUTInject and forward successful srcaddr: 198.51.100.38, dstaddr:
198.51.100.55

  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 198.51.100.38:22 198.51.100.55:52774 seq 3052140910
```

```
OPTS 4 ACK 2346709419 SYN  WIN 4128
Summary
  Input     : INJ.2
  Output    : GigabitEthernet1
  State     : FWD
  Timestamp
    Start   : 490928006866 ns (06/29/2020 13:31:30.807879 UTC)
    Stop    : 490928038567 ns (06/29/2020 13:31:30.807911 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : internal0/0/rp:0
    Output      : <unknown>
    Source      : 172.18.124.38
    Destination : 172.18.124.55
    Protocol    : 6 (TCP)
      SrcPort   : 22
      DstPort   : 52774
  Feature: IPSec
    Result    : IPSEC_RESULT_DENY
    Action    : SEND_CLEAR
    SA Handle : 0
    Peer Addr : 55.124.18.172
    Local Addr: 38.124.18.172


Router#
```

Additional References

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 this URL: {start hypertext}http://www.cisco.com/go/mibs{end hypertext}

RFCs

RFC	Title
None	—

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.	{start hypertext}http://www.cisco.com/cisco/web/support/index.html{end hypertext}

Feature Information for Packet Trace

{start cross reference}Table 21-4{end cross reference} lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to{start hypertext} http://www.cisco.com/go/cfn{end hypertext}. An account on Cisco.com is not required.

Note {start cross reference}Table 21-4{end cross reference} lists only the software releases that support a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 26: Feature Information for Packet Trace

Feature Name	Releases	Feature Information
Packet Trace	Cisco IOS XE 3.10S	The Packet Trace feature provides information about how data packets are processed by the Cisco IOS XE software. In Cisco IOS XE Release 3.10S, this feature was introduced. The following commands were introduced or modified: • **debug platform packet-trace packet** *pkt-num* [**fia-trace** | **summary-only**] [**data-size** *data-size*] [**circular**] • **debug platform packet-trace copy packet** {**input** | **output** | **both**} [**size** *num-bytes*] [**L2** | **L3** | **L4**] • **show platform packet-trace** {**configuration** | **statistics** | **summary** | **packet** {**all** | *pkt-num*}}
	Cisco IOS XE 3.11S	In Cisco IOS XE Release 3.11S, this feature was enhanced to include the following features: • Matched versus traced statistics. • Trace stop timestamp in addition to trace start timestamp. The following commands were introduced or modified: • **debug platform packet-trace drop** [**code** *drop-num*] • **show platform packet-trace packet** {**all** | *pkt-num*} [*decode*]
	Cisco IOS XE Denali 16.3.1	In Cisco IOS XE Denali 16.3.1, this feature was enhanced to include Layer3 packet tracing along with IOSd. The following commands were introduced or modified: **debug platform packet-trace punt**.
	Cisco IOS XE Amsterdam 17.3.1	The output of the **show platform packet-trace** command now includes additional trace information for packets either originated from IOSd or destined to IOSd or other BinOS processes.

CHAPTER 17

Environmental Monitoring and PoE Management

The Cisco 4000 series Integrated Services routers have hardware and software features that periodically monitor the router's environment. For more information, see the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

This chapter provides information on the environmental monitoring features on your router that allow you to monitor critical events and generate statistical reports on the status of various router components and, includes the following sections:

Environmental Monitoring

The router provides a robust environment-monitoring system with several sensors that monitor the system temperatures. Microprocessors generate interrupts to the HOST CPU for critical events and generate a periodic status and statistics report. The following are some of the key functions of the environmental monitoring system:

- Monitoring temperature of CPUs, motherboard, and midplane
- Monitoring fan speed
- Recording abnormal events and generating notifications
- Monitoring Simple Network Management Protocol (SNMP) traps
- Generating and collecting Onboard Failure Logging (OBFL) data
- Sending call home event notifications
- Logging system error messages
- Displaying present settings and status

Environmental Monitoring and Reporting Functions

Monitoring and reporting functions allow you to maintain normal system operation by identifying and resolving adverse conditions prior to loss of operation.

Environmental Monitoring Functions

Environmental monitoring functions use sensors to monitor the temperature of the cooling air as it moves through the chassis.

The local power supplies provide the ability to monitor:

- Input and output current
- Output voltage
- Input and output power
- Temperature
- Fan speed

The router is expected to meet the following environmental operating conditions:

- Operating Temperature Nominal—32°F to 104°F (0°C to 40°C)
- Operating Humidity Nominal—10% to 85% RH noncondensing
- Operating Humidity Short Term—10% to 85% RH noncondensing
- Operating Altitude—Sea level 0 ft to 10,000 ft (0 to 3000 m)
- AC Input Range—85 to 264 VAC

In addition, each power supply monitors its internal temperature and voltage. A power supply is either within tolerance (normal) or out of tolerance (critical). If an internal power supply's temperature or voltage reaches a critical level, the power supply shuts down without any interaction with the system processor.

The following table displays the levels of status conditions used by the environmental monitoring system.

Table 27: Levels of Status Conditions Used by the Environmental Monitoring System

Status Level	Description
Normal	All monitored parameters are within normal tolerance.
Warning	The system has exceeded a specified threshold. The system continues to operate, but operator action is recommended to bring the system back to a normal state.

Status Level	Description
Critical	An out-of-tolerance temperature or voltage condition exists. Although the system continues to operate, it is approaching shutdown. Immediate operator action is required.

The environmental monitoring system sends system messages to the console, for example, when the conditions described here are met:

Fan Failure

When the system power is on, all the fans should be operational. Although the system continues to operate if a fan fails, the system displays the following message:

```
%IOSXE_PEM-3-FANFAIL: The fan in slot 2/0 is encountering a failure condition
```

Sensors Out of Range

When sensors are out of range, the system displays the following message:

```
%ENVIRONMENTAL-1-ALERT: V: 1.0v PCH, Location: R0, State: Warning, Reading: 1102 mV
```

```
%ENVIRONMENTAL-1-ALERT: V: PEM Out, Location: P1, State: Warning, Reading: 0 mV
```

```
%ENVIRONMENTAL-1-ALERT: Temp: Temp 3, Location R0, State : Warning, Reading : 90C
```

Fan Tray (Slot P2) Removed

When the fan tray for slot P2 is removed, the system displays the following message:

```
%IOSXE_PEM-6-REMPEM_FM: PEM/FM slot P2 removed
```

Fan Tray (Slot P2) Reinserted

When the fan tray for slot P2 is reinserted, the system displays the following message:

```
%IOSXE_PEM-6-INSPEM_FM: PEM/FM slot P2 inserted
```

Fan Tray (Slot 2) is Working Properly

When the fan tray for slot 2 is functioning properly, the system displays the following message:

```
%IOSXE_PEM-6-PEMOK: The PEM in slot P2 is functioning properly
```

Fan 0 in Slot 2 (Fan Tray) is Not Working

When Fan 0 in the fan tray of slot 2 is not functioning properly, the system displays the following message:

```
%IOSXE_PEM-3-FANFAIL: The fan in slot 2/0 is encountering a failure condition
```

Fan 0 in Slot 2 (Fan Tray) is Working Properly

When Fan 0 in the fan tray of slot 2 is functioning properly, the system displays the following message:

```
%IOSXE_PEM-6-FANOK: The fan in slot 2/0 is functioning properly
```

Main Power Supply in Slot 1 is Powered Off

When the main power supply in slot 1 is powered off, the system displays the following message:

```
%IOSXE_PEM-3-PEMFAIL: The PEM in slot 1 is switched off or encountering a
failure condition.
```

Main Power Supply is Inserted in Slot 1

When the main power supply is inserted in slot 1, the system displays the following messages:

```
%IOSXE_PEM-6-INSPEM_FM: PEM/FM slot P1 inserted
%IOSXE_PEM-6-PEMOK: The PEM in slot 1 is functioning properly
```

Temperature and Voltage Exceed Max/Min Thresholds

The following example shows the warning messages indicating the maximum and minimum thresholds of the temperature or voltage:

```
Warnings :
--------
For all the temperature sensors (name starting with “Temp:”) above,
the critical warning threshold is 100C (100C and higher)
the warning threshold is 80C (range from 80C to 99C)
the low warning threshold is 1C (range from -inf to 1C).

For all voltage sensors (names starting with "V:"),
the high warning threshold starts at that voltage +10%. (voltage + 10% is warning)
the low warning threshold starts at the voltage -10%. (voltage - 10% is warning)
```

Environmental Reporting Functions

You can retrieve and display environmental status reports using the following commands:

- **debug environment**
- **debug platform software cman env monitor polling**
- **debug ilpower**
- **debug power** [**inline** | **main**]
- **show diag all eeprom**
- **show diag slot R0 eeprom detail**
- **show environment**
- **show environment all**
- **show inventory**
- **show platform all**
- **show platform diag**
- **show platform software status control-processor**
- **show version**
- **show power**
- **show power inline**

These commands show the current values of parameters such as temperature and voltage.

The environmental monitoring system updates the values of these parameters every 60 seconds. Brief examples of these commands are shown below:

debug environment: Example

```
Router# debug environment location P0
Environmental sensor Temp: Temp 1 P0 debugging is on
Environmental sensor Temp: Temp 2 P0 debugging is on
Environmental sensor Temp: Temp 3 P0 debugging is on
Environmental sensor V: PEM Out P0 debugging is on
Environmental sensor I: PEM In P0 debugging is on
Environmental sensor I: PEM Out P0 debugging is on
Environmental sensor W: In pwr P0 debugging is on
Environmental sensor W: Out pwr P0 debugging is on
Environmental sensor RPM: fan0 P0 debugging is on

*Sep 12 00:45:13.956: Sensor: Temp: Temp 1 P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=29
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: Temp: Temp 1 P0 State=Normal Reading=29
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
*Sep 12 00:45:13.956: Sensor: Temp: Temp 2 P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=33
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: Temp: Temp 2 P0 State=Normal Reading=34
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
*Sep 12 00:45:13.956: Sensor: Temp: Temp 3 P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=34
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: Temp: Temp 3 P0 State=Normal Reading=35
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
*Sep 12 00:45:13.956: Sensor: V: PEM Out P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=12709
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: V: PEM Out P0 State=Normal Reading=12724
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
*Sep 12 00:45:13.956: Sensor: I: PEM In P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=1
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: I: PEM In P0 State=Normal Reading=1
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
*Sep 12 00:45:13.956: Sensor: I: PEM Out P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=4
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: I: PEM Out P0 State=Normal Reading=4
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
*Sep 12 00:45:13.956: Sensor: W: In pwr P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=92
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: W: In pwr P0 State=Normal Reading=92
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
*Sep 12 00:45:13.956: Sensor: W: Out pwr P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=46
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: W: Out pwr P0 State=Normal Reading=46
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
```

```
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
*Sep 12 00:45:13.956: Sensor: RPM: fan0 P0, In queue 1
*Sep 12 00:45:13.956: State=Normal Reading=3192
*Sep 12 00:45:13.956: Rotation count=0 Poll period=60000
*Sep 12 00:45:13.956: Sensor: RPM: fan0 P0 State=Normal Reading=3180
*Sep 12 00:45:13.956: Inserting into queue 1 on spoke 173.
*Sep 12 00:45:13.956: Rotation count=60 Displacement=0
```

debug platform software cman env monitor polling: Example

```
Router# debug platform software cman env monitor polling
platform software cman env monitor polling debugging is on
Router#
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback Temp: Temp 1, P0, 29
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback Temp: Temp 2, P0, 34
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback Temp: Temp 3, P0, 35
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback V: PEM Out, P0, 12709
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback I: PEM In, P0, 1
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback I: PEM Out, P0, 4
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback W: In pwr, P0, 93
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback W: Out pwr, P0, 48
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback RPM: fan0, P0, 3192
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback Temp: Temp 1, P1, 33
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback Temp: Temp 2, P1, 32
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback Temp: Temp 3, P1, 36
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback V: PEM Out, P1, 12666
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback I: PEM In, P1, 1
*Sep 12 00:46:13.962: IOS-RP-ENVMON: sensor READ callback I: PEM Out, P1, 4
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback W: In pwr, P1, 55
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback W: Out pwr, P1, 46
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback RPM: fan0, P1, 2892
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback RPM: fan0, P2, 4894
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback RPM: fan1, P2, 4790
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback RPM: fan2, P2, 5025
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback RPM: fan3, P2, 5001
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback W: fan pwr, P2, 8
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback Temp: Inlet 1, R0, 25
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback Temp: Inlet 2, R0, 28
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback Temp: Outlet 1, R0, 30
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback Temp: Outlet 2, R0, 35
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 12v, R0, 12735
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 5v, R0, 5125
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 3.3v, R0, 3352
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.05v, R0, 1052
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 2.5v, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.8v, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.2v, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.15v, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.1v, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.0v, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.8v PCH, R0, 1787
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.5v PCH, R0, 1516
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.5v CPUC, R0, 1526
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.5v CPUI, R0, 1529
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.0v PCH, R0, 1009
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 1.5v QLM, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: VCore, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: VTT, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 0.75v CPUI, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback V: 0.75v CPUC, R0, 0
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback I: 12v, R0, 7
*Sep 12 00:46:13.963: IOS-RP-ENVMON: sensor READ callback W: pwr, R0, 81
```

debug ilpower: Example

```
Router# debug ilpower ?
cdp ILPOWER CDP messages
controller ILPOWER controller
event ILPOWER event
ha ILPOWER High-Availability
port ILPOWER port management
powerman ILPOWER powerman
registries ILPOWER registries
scp ILPOWER SCP messages
```

debug power [inline|main]: Example

In this example, there is one 1000W power supply and one 450W power supply. Inline and main power output is shown.

```
Router# debug power ?
inline ILPM inline power related
main Main power related
<cr>
Router# debug power
POWER all debug debugging is on

Router# show debugging | include POWER
POWER:
POWER main debugging is on
POWER inline debugging is on
Router#
..
*Jan 21 01:29:40.786: %ENVIRONMENTAL-6-NOTICE: V: PEM Out, Location: P1, State: Warning,
Reading: 0 mV
*Jan 21 01:29:43.968: %IOSXE_PEM-6-PEMOK: The PEM in slot P1 is functioning properly
*Jan 21 01:29:43.968: %PLATFORM_POWER-6-MODEMATCH: Main power is in Boost mode
*Jan 21 01:29:43.968: Power M: Received Msg for 12V/Main, total power 1450, Run same as cfg
 Yes
*Jan 21 01:29:43.968: Power M: Received Msg for POE/ILPM, total power 500, Run same as cfg
 No
*Jan 21 01:29:43.968: Power I: Updating pool power is 500 watts
*Jan 21 01:29:43.968: Power I: Intimating modules of total power 500 watts
*Jan 21 01:29:46.488: Power M: Received Msg for 12V/Main, total power 1450, Run same as cfg
 Yes
*Jan 21 01:29:46.488: Power M: Received Msg for POE/ILPM, total power 500, Run same as cfg
 No
*Jan 21 01:29:46.488: Power I: Updating pool power is 500 watts
*Jan 21 01:29:46.488: Power I: Intimating modules of total power 500 watts
Router#
```

show diag all eeprom: Example

```
Router# show diag all eeprom
MIDPLANE EEPROM data:

Product Identifier (PID) : ISR4451/K9
Version Identifier (VID) : V01
PCB Serial Number : FOC15507S9K
Hardware Revision : 1.0
Asset ID : P1B-R2C-CP1.0
CLEI Code : TDBTDBTDBT
```

```
Power/Fan Module P0 EEPROM data:

Product Identifier (PID) : XXX-XXXX-XX
Version Identifier (VID) : XXX
PCB Serial Number : DCA1547X047
CLEI Code : 0000000000
Power/Fan Module P1 EEPROM data:

Product Identifier (PID) : XXX-XXXX-XX
Version Identifier (VID) : XXX
PCB Serial Number : DCA1533X022
CLEI Code : 0000000000
Power/Fan Module P2 EEPROM data is not initialized

Internal PoE is not present
Slot R0 EEPROM data:

Product Identifier (PID) : ISR4451/K9
Version Identifier (VID) : V01
PCB Serial Number : FOC15507S9K
Hardware Revision : 1.0
CLEI Code : TDBTDBTDBT
Slot F0 EEPROM data:

Product Identifier (PID) : ISR4451-FP
Version Identifier (VID) : V00
PCB Serial Number : FP123456789
Hardware Revision : 4.1
Slot 0 EEPROM data:

Product Identifier (PID) : ISR4451/K9
Version Identifier (VID) : V01
PCB Serial Number : FOC15507S9K
Hardware Revision : 1.0
CLEI Code : TDBTDBTDBT
Slot 1 EEPROM data:

Product Identifier (PID) : ISR4451/K9
Version Identifier (VID) : V01
PCB Serial Number : FOC15507S9K
Hardware Revision : 1.0
CLEI Code : TDBTDBTDBT
Slot 2 EEPROM data:

Product Identifier (PID) : ISR4451/K9
Version Identifier (VID) : V01
PCB Serial Number : FOC15507S9K
Hardware Revision : 1.0
CLEI Code : TDBTDBTDBT
SPA EEPROM data for subslot 0/0:

Product Identifier (PID) : ISR441-4X1GE
Version Identifier (VID) : V01
PCB Serial Number : JAB092709EL
Top Assy. Part Number : 68-2236-01
Top Assy. Revision : A0
Hardware Revision : 2.2
CLEI Code : CNUIAHSAAA
SPA EEPROM data for subslot 0/1 is not available

SPA EEPROM data for subslot 0/2 is not available

SPA EEPROM data for subslot 0/3 is not available
```

```
SPA EEPROM data for subslot 0/4 is not available

SPA EEPROM data for subslot 1/0 is not available

SPA EEPROM data for subslot 1/1 is not available

SPA EEPROM data for subslot 1/2 is not available

SPA EEPROM data for subslot 1/3 is not available

SPA EEPROM data for subslot 1/4 is not available

SPA EEPROM data for subslot 2/0 is not available

SPA EEPROM data for subslot 2/1 is not available

SPA EEPROM data for subslot 2/2 is not available

SPA EEPROM data for subslot 2/3 is not available
SPA EEPROM data for subslot 2/4 is not available
```

show environment: Example

In this example, note the output for the slots POE0 and POE1. Cisco IOS XE 3.10 and higher supports an external PoE module.

```
Router# show environment

Number of Critical alarms: 0
Number of Major alarms: 0
Number of Minor alarms: 0

Slot Sensor Current State Reading
---- ------ ------------- -------
P0 Temp: Temp 1 Normal 28 Celsius
P0 Temp: Temp 2 Normal 43 Celsius
P0 Temp: Temp 3 Normal 44 Celsius
P0 V: PEM Out Normal 12404 mV
P0 I: PEM In Normal 1 A
P0 I: PEM Out Normal 7 A
P0 P: In pwr Normal 106 Watts
P0 P: Out pwr Normal 87 Watts
P0 RPM: fan0 Normal 2952 RPM
P2 RPM: fan0 Normal 4421 RPM
P2 RPM: fan1 Normal 4394 RPM
P2 RPM: fan2 Normal 4433 RPM
P2 RPM: fan3 Normal 4410 RPM
P2 P: pwr Normal 6 Watts
POE0 Temp: Temp 1 Normal 44 Celsius
POE0 I: 12v In Normal 2 A
POE0 V: 12v In Normal 12473 mV
POE0 P: In pwr Normal 25 Watts
POE1 Temp: Temp 1 Normal 40 Celsius
POE1 I: 12v In Normal 2 mA
POE1 V: 12v In Normal 12473 mV
POE1 P: In pwr Normal 20 Watts
R0 Temp: Inlet 1 Normal 24 Celsius
R0 Temp: Inlet 2 Normal 26 Celsius
R0 Temp: Outlet 1 Normal 33 Celsius
R0 Temp: Outlet 2 Normal 32 Celsius
R0 Temp: core-B Normal 43 Celsius
```

```
R0 Temp: core-C Normal 38 Celsius
R0 V: 12v Normal 12355 mV
R0 V: 5v Normal 5090 mV
R0 V: 3.3v Normal 3331 mV
R0 V: 3.0v Normal 2998 mV
R0 V: 2.5v Normal 2436 mV
R0 V: 1.05v Normal 1049 mV
R0 V: 1.8v Normal 1798 mV
R0 V: 1.2v Normal 1234 mV
R0 V: Vcore-C Normal 1155 mV
R0 V: 1.1v Normal 1104 mV
R0 V: 1.0v Normal 1012 mV
R0 V: 1.8v-A Normal 1782 mV
R0 V: 1.5v-A Normal 1505 mV
R0 V: 1.5v-C1 Normal 1516 mV
R0 V: 1.5v-B Normal 1511 mV
R0 V: Vcore-A Normal 1099 mV
R0 V: 1.5v-C2 Normal 1492 mV
R0 V: Vcore-B1 Normal 891 mV
R0 V: Vcore-B2 Normal 904 mV
R0 V: 0.75v-B Normal 754 mV
R0 V: 0.75v-C Normal 759 mV
R0 I: 12v Normal 8 A
R0 P: pwr Normal 86 Watts
0/1 P: pwr Normal 5 Watts
P1 Temp: Temp 1 Normal 30 Celsius
P1 Temp: Temp 2 Normal 38 Celsius
P1 Temp: Temp 3 Normal 39 Celsius
P1 V: PEM Out Normal 12404 mV
P1 I: PEM In Normal 1 A
P1 I: PEM Out Normal 6 A
P1 P: In pwr Normal 86 Watts
P1 P: Out pwr Normal 68 Watts
P1 RPM: fan0 Normal 2940 RPM
```

show environment all: Example

```
Router# show environment all
Sensor List: Environmental Monitoring
Sensor Location State Reading
Temp: Temp 1 P0 Normal 29 Celsius
Temp: Temp 2 P0 Normal 43 Celsius
Temp: Temp 3 P0 Normal 44 Celsius
V: PEM Out P0 Normal 12404 mV
I: PEM In P0 Normal 1 A
I: PEM Out P0 Normal 8 A
P: In pwr P0 Normal 111 Watts
P: Out pwr P0 Normal 91 Watts
RPM: fan0 P0 Normal 2940 RPM
RPM: fan0 P2 Normal 4419 RPM
RPM: fan1 P2 Normal 4395 RPM
RPM: fan2 P2 Normal 4426 RPM
RPM: fan3 P2 Normal 4412 RPM
P: pwr P2 Normal 6 Watts
Temp: Temp 1 POE0 Normal 44 Celsius
I: 12v In POE0 Normal 2 A
V: 12v In POE0 Normal 12473 mV
P: In pwr POE0 Normal 25 Watts
Temp: Temp 1 POE1 Normal 40 Celsius
I: 12v In POE1 Normal 2 mA
V: 12v In POE1 Normal 12473 mV
```

```
P: In pwr POE1 Normal 20 Watts
Temp: Inlet 1 R0 Normal 24 Celsius
Temp: Inlet 2 R0 Normal 27 Celsius
Temp: Outlet 1 R0 Normal 33 Celsius
Temp: Outlet 2 R0 Normal 32 Celsius
Temp: core-B R0 Normal 49 Celsius
Temp: core-C R0 Normal 37 Celsius
V: 12v R0 Normal 12355 mV
V: 5v R0 Normal 5084 mV
V: 3.3v R0 Normal 3331 mV
V: 3.0v R0 Normal 2998 mV
V: 2.5v R0 Normal 2433 mV
V: 1.05v R0 Normal 1052 mV
V: 1.8v R0 Normal 1798 mV
V: 1.2v R0 Normal 1226 mV
V: Vcore-C R0 Normal 1155 mV
V: 1.1v R0 Normal 1104 mV
V: 1.0v R0 Normal 1015 mV
V: 1.8v-A R0 Normal 1782 mV
V: 1.5v-A R0 Normal 1508 mV
V: 1.5v-C1 R0 Normal 1513 mV
V: 1.5v-B R0 Normal 1516 mV
V: Vcore-A R0 Normal 1099 mV
V: 1.5v-C2 R0 Normal 1492 mV
V: Vcore-B1 R0 Normal 1031 mV
V: Vcore-B2 R0 Normal 901 mV
V: 0.75v-B R0 Normal 754 mV
V: 0.75v-C R0 Normal 754 mV
I: 12v R0 Normal 8 A
P: pwr R0 Normal 97 Watts
P: pwr 0/1 Normal 5 Watts
Temp: Temp 1 P1 Normal 30 Celsius
Temp: Temp 2 P1 Normal 39 Celsius
Temp: Temp 3 P1 Normal 39 Celsius
V: PEM Out P1 Normal 12404 mV
I: PEM In P1 Normal 1 A
I: PEM Out P1 Normal 6 A
P: In pwr P1 Normal 87 Watts
P: Out pwr P1 Normal 66 Watts
RPM: fan0 P1 Normal 2940 RPM
```

show inventory: Example

```
Router# show inventory
NAME: "Chassis", DESCR: "Cisco ISR4451 Chassis"
PID: ISR4451/K9 , VID: V01, SN: FGL160110QZ

NAME: "Power Supply Module 0", DESCR: "450W AC Power Supply for Cisco ISR4450"
PID: XXX-XXXX-XX , VID: XXX, SN: DCA1547X047

NAME: "Power Supply Module 1", DESCR: "450W AC Power Supply for Cisco ISR4450"
PID: XXX-XXXX-XX , VID: XXX, SN: DCA1614Y022

NAME: "Fan Tray", DESCR: "Cisco ISR4450 Fan Assembly"
PID: ACS-4450-FANASSY , VID: , SN:

NAME: "POE Module 0", DESCR: "Single POE for Cisco ISR4451"
PID: PWR-POE-4400 , VID: , SN: FHH1638P00E

NAME: "POE Module 1", DESCR: "Single POE for Cisco ISR4451"
PID: PWR-POE-4400 , VID: , SN: FHH1638P00G
```

```
NAME: "GE-POE Module", DESCR: "POE Module for On Board GE for Cisco ISR4400"
PID: 800G2-POE-2 , VID: V01, SN: FOC151849W9

NAME: "module 0", DESCR: "Cisco ISR4451 Built-In NIM controller"
PID: ISR4451/K9 , VID: , SN:
NAME: "NIM subslot 0/2", DESCR: " NIM-4MFT-T1/E1 - T1/E1 Serial Module"
PID: NIM-4MFT-T1/E1 , VID: V01, SN: FOC16254E6W

NAME: "NIM subslot 0/3", DESCR: "NIM SSD Module"
PID: NIM-SSD , VID: V01, SN: FHH16510032

NAME: "NIM subslot 0/0", DESCR: "Front Panel 4 ports Gigabitethernet Module"
PID: ISR4451-X-4x1GE , VID: V01, SN: JAB092709EL

NAME: "module 1", DESCR: "Cisco ISR4451 Built-In SM controller"
PID: ISR4451/K9 , VID: , SN:

NAME: "SM subslot 1/0", DESCR: "SM-X-1T3/E3 - Clear T3/E3 Serial Module"
PID: SM-X-1T3/E3 , VID: V01, SN: FOC164750RG

NAME: "module 2", DESCR: "Cisco ISR4451 Built-In SM controller"
PID: ISR4451/K9 , VID: , SN:

NAME: "SM subslot 2/0", DESCR: "SM-ES3X-24-P: EtherSwitch SM L3 + PoEPlus + MACSec + 24
10/100/1000"
PID: SM-ES3X-24-P , VID: V01, SN: FHH1629007C

NAME: "module R0", DESCR: "Cisco ISR4451 Route Processor"
PID: ISR4451/K9 , VID: V01, SN: FOC15507S95

NAME: "module F0", DESCR: "Cisco ISR4451 Forwarding Processor"
PID: ISR4451/K9 , VID: , SN:
```

Note Cisco ISR 4321 does not display the serial numbers of power supply and fan tray with the **show inventory** command.

show platform: Example

```
Router# show platform
Chassis type: ISR4451/K9

Slot Type State Insert time (ago)
--------- ------------------- --------------------- -----------------
0 ISR4451/K9 ok 3d11h
0/0 ISR4451-X-4x1GE ok 3d11h
0/2 NIM-4MFT-T1/E1 ok 3d11h
0/3 NIM-SSD ok 3d11h
1 ISR4451/K9 ok 3d11h
1/0 SM-X-1T3/E3 ok 3d11h
2 ISR4451/K9 ok 3d11h
2/0 SM-ES3X-24-P ok 3d11h
R0 ISR4451/K9 ok, active 3d11h
F0 ISR4451/K9 ok, active 3d11h
P0 XXX-XXXX-XX ok 3d11h
P1 XXX-XXXX-XX ok 3d11h
P2 ACS-4450-FANASSY ok 3d11h
POE0 PWR-POE-4400 ok 3d11h
```

```
POE1 PWR-POE-4400 ok 3d11h
GE-POE 800G2-POE-2 ok 3d11h
```

show platform diag: Example

```
Router# show platform diag
Chassis type: ISR4451/K9

Slot: 0, ISR4451/K9
Running state : ok
Internal state : online
Internal operational state : ok
Physical insert detect time : 00:01:04 (3d10h ago)
Software declared up time : 00:01:43 (3d10h ago)
CPLD version : 12121625
Firmware version : 15.3(1r)S

Sub-slot: 0/0, ISR4451-X-4x1GE
Operational status : ok
Internal state : inserted
Physical insert detect time : 00:03:03 (3d10h ago)
Logical insert detect time : 00:03:03 (3d10h ago)

Sub-slot: 0/2, NIM-4MFT-T1/E1
Operational status : ok
Internal state : inserted
Physical insert detect time : 00:03:03 (3d10h ago)
Logical insert detect time : 00:03:03 (3d10h ago)

Sub-slot: 0/3, NIM-SSD
Operational status : ok
Internal state : inserted
Physical insert detect time : 00:03:03 (3d10h ago)
Logical insert detect time : 00:03:03 (3d10h ago)

Slot: 1, ISR4451/K9
Running state : ok
Internal state : online
Internal operational state : ok
Physical insert detect time : 00:01:04 (3d10h ago)
Software declared up time : 00:01:44 (3d10h ago)
CPLD version : 12121625
Firmware version : 15.3(1r)S

Sub-slot: 1/0, SM-X-1T3/E3
Operational status : ok
Internal state : inserted
Physical insert detect time : 00:03:03 (3d10h ago)
Logical insert detect time : 00:03:03 (3d10h ago)

Slot: 2, ISR4451/K9
Running state : ok
Internal state : online
Internal operational state : ok
Physical insert detect time : 00:01:04 (3d10h ago)
Software declared up time : 00:01:45 (3d10h ago)
CPLD version : 12121625
Firmware version : 15.3(1r)S

Sub-slot: 2/0, SM-ES3X-24-P
Operational status : ok
```

```
Internal state : inserted
Physical insert detect time : 00:03:03 (3d10h ago)
Logical insert detect time : 00:03:03 (3d10h ago)

Slot: R0, ISR4451/K9
Running state : ok, active
Internal state : online
Internal operational state : ok
Physical insert detect time : 00:01:04 (3d10h ago)
Software declared up time : 00:01:04 (3d10h ago)
CPLD version : 12121625
Firmware version : 15.3(1r)S

Slot: F0, ISR4451/K9
Running state : ok, active
Internal state : online
Internal operational state : ok
Physical insert detect time : 00:01:04 (3d10h ago)
Software declared up time : 00:02:39 (3d10h ago)
Hardware ready signal time : 00:00:00 (never ago)
Packet ready signal time : 00:02:48 (3d10h ago)
CPLD version : 12121625
Firmware version : 15.3(1r)S

Slot: P0, XXX-XXXX-XX
State : ok
Physical insert detect time : 00:01:29 (3d10h ago)

Slot: P1, XXX-XXXX-XX
State : ok
Physical insert detect time : 00:01:29 (3d10h ago)

Slot: P2, ACS-4450-FANASSY
State : ok
Physical insert detect time : 00:01:29 (3d10h ago)

Slot: POE0, PWR-POE-4451
State : ok
Physical insert detect time : 00:01:29 (3d10h ago)

Slot: POE1, PWR-POE-4451
State : ok
Physical insert detect time : 00:01:29 (3d10h ago)

Slot: GE-POE, 800G2-POE-2
State : ok
Physical insert detect time : 00:01:29 (3d10h ago)
```

show platform software status control-processor: Example

```
Router# show platform software status control-processor
RP0: online, statistics updated 2 seconds ago
Load Average: health unknown
1-Min: 0.13, status: health unknown, under
5-Min: 0.07, status: health unknown, under
15-Min: 0.06, status: health unknown, under
Memory (kb): healthy
Total: 3971244
Used: 2965856 (75%)
Free: 1005388 (25%)
Committed: 2460492 (62%), status: health unknown, under 0%
```

```
Per-core Statistics
CPU0: CPU Utilization (percentage of time spent)
User: 1.00, System: 2.90, Nice: 0.00, Idle: 96.00
IRQ: 0.10, SIRQ: 0.00, IOwait: 0.00
CPU1: CPU Utilization (percentage of time spent)
User: 10.71, System: 29.22, Nice: 0.00, Idle: 60.06
IRQ: 0.00, SIRQ: 0.00, IOwait: 0.00
CPU2: CPU Utilization (percentage of time spent)
User: 0.80, System: 1.30, Nice: 0.00, Idle: 97.90
IRQ: 0.00, SIRQ: 0.00, IOwait: 0.00
CPU3: CPU Utilization (percentage of time spent)
User: 10.61, System: 34.03, Nice: 0.00, Idle: 55.25
IRQ: 0.00, SIRQ: 0.10, IOwait: 0.00
CPU4: CPU Utilization (percentage of time spent)
User: 0.60, System: 1.20, Nice: 0.00, Idle: 98.20
IRQ: 0.00, SIRQ: 0.00, IOwait: 0.00
CPU5: CPU Utilization (percentage of time spent)
User: 13.18, System: 35.46, Nice: 0.00, Idle: 51.24
IRQ: 0.00, SIRQ: 0.09, IOwait: 0.00
CPU6: CPU Utilization (percentage of time spent)
User: 0.80, System: 2.40, Nice: 0.00, Idle: 96.80
IRQ: 0.00, SIRQ: 0.00, IOwait: 0.00
CPU7: CPU Utilization (percentage of time spent)
User: 10.41, System: 33.63, Nice: 0.00, Idle: 55.85
IRQ: 0.00, SIRQ: 0.10, IOwait: 0.00
```

show diag slot RO eeprom detail: Example

```
Router# show diag slot R0 eeprom detail
Slot R0 EEPROM data:

EEPROM version : 4
Compatible Type : 0xFF
PCB Serial Number : FHH153900AU
Controller Type : 1902
Hardware Revision : 0.0
PCB Part Number : 73-13854-01
Top Assy. Part Number : 800-36894-01
Board Revision : 01
Deviation Number : 122081
Fab Version : 01
Product Identifier (PID) : CISCO------<0A>
Version Identifier (VID) : V01<0A>
Chassis Serial Number : FHH1539P00Q
Chassis MAC Address : 0000.0000.0000
MAC Address block size : 96
Asset ID : REV1B<0A>
Asset ID :
```

show version: Example

```
Router# show version
Cisco IOS XE Software, Version 03.13.00.S - Standard Support Release
Cisco IOS Software, ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Version 15.4(3)S, RELEASE
 SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2014 by Cisco Systems, Inc.
Compiled Tue 27-May-14 05:36 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2014 by cisco Systems, Inc.
```

```
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

Router uptime is 2 hours, 19 minutes
Uptime for this control processor is 2 hours, 22 minutes
System returned to ROM by reload
System image file is "tftp: isr4400-universalk9.03.13.00.S.154-3.S-std.SPA.bin"
Last reload reason: Reload Command


This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

Technology Package License Information:

-----------------------------------------------------------------
Technology    Technology-package           Technology-package
                   Current        Type             Next reboot
------------------------------------------------------------------
appx               None            None                None
uc                 None            None                None
security           None            None                None
ipbase             ipbasek9       Permanent            ipbasek9

cisco 4451 ISR processor with 1213154K/6147K bytes of memory.
Processor board ID FHH1539P00Q
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
3391455K bytes of Compact flash at bootflash:.

Configuration register is 0x0”
```

Configuring Power Supply Mode

You can configure the power supplies of both the router and a connected Power over Ethernet (PoE) module.

• Configuring the Router Power Supply Mode, on page 243

Configuring the Router Power Supply Mode

Configure the main power supply on the router using the **power main redundant** command:

- **power main redundant**—Sets the main power supply in redundant mode.
- **no power main redundant**—Sets the main power supply in boost mode.

Note The default mode for the router power supply is redundant mode.

Configuring the External PoE Service Module Power Supply Mode

Configure the power supply of an external PoE service module using the **power inline redundant** command:

- **power inline redundant**—Sets the external PoE service module power supply in redundant mode.
- **no power inline redundant**—Sets the external PoE service module power supply in boost mode.

Note The default mode for the external PoE service module power supply is redundant mode.

The **show power** command shows whether boost or redundant mode is configured and whether this mode is currently running on the system.

Examples for Configuring Power Supply Mode

Example—Configured Mode of Boost for Main PSU and PoE Module

In this example, the **show power** command shows the configured mode as `Boost`, which is also the current runtime state. The `Main PSU` shows information about the main power supply. The `POE Module` shows information about the inline/PoE power. In this example, the current run-time state for the main power supply is the same as the configured state (`Boost` mode).

```
Router# show power
Main PSU :
Configured Mode : Boost
Current runtime state same : Yes
Total power available : 2000 Watts
POE Module :
Configured Mode : Boost
Current runtime state same : Yes
Total power available : 1000 Watts
Router#
```

Example—Configured Mode of Boost for Main PSU and PoE Module

In this example, the **show power** command shows the power supplies that are present in the device. The Main PSU and POE Module are configured to the `Boost` mode, which differs from the current runtime state. The current runtime state is the `Redundant` mode. A likely explanation for this is that there is only one main power supply present in the router. See mode example 4 in the table titled "Modes of Operation" in Available PoE Power, on page 245.

You can enter the **show platform** command to show the power supplies that are present in the device.

```
Router# show power
Main PSU :
Configured Mode : Boost
Current runtime state same : No
Total power available : 1000 Watts
POE Module :
Configured Mode : Boost
Current runtime state same : No
Total power available : 500 Watts
Router#
```

Example—Configured Mode of Redundant for Main PSU and PoE Module

In this example, the **show power** command shows the configured mode is `Redundant` for both the main and inline power. The system has one 450 W and one 100 W power supply.

```
Router# show power
Main PSU :
Configured Mode : Redundant
Current runtime state same : Yes
Total power available : 450 Watts
POE Module :
Configured Mode : Redundant
Current runtime state same : No
Total power available : 0 Watts
Router#
```

Example—Configured Mode of Boost for Main Power

In this example, the main power is configured to be in `boost` mode by using the **no** form of the **power main redundant** command. This sets the main power to `boost` mode with 1450 W and inline power to `redundant` mode with 500 W.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# no power main redundant
Router(config)#
*Jan 31 03:35:22.284: %PLATFORM_POWER-6-MODEMATCH: Inline power is in Redundant mode
Router(config)#
Router(config)# exit
Router#
*Jan 31 03:36:13.111: %SYS-5-CONFIG_I: Configured from console by console
Router# show power
Main PSU :
Configured Mode : Boost
Current runtime state same : Yes
Total power available : 1450 Watts
POE Module :
```

```
Configured Mode : Redundant
Current runtime state same : Yes
Total power available : 500 Watts
Router#
```

Example—Configured Mode of Boost for PoE Power

In this example, an attempt is made to configure the inline power in boost mode by using the **no** form of the **power inline redundant** command. The inline power mode is **not** changed to boost mode because that would require a total power available in redundant mode of 1000 W. The inline power mode is redundant and is shown by the following values for the PoE Module:

- `Configured Mode : Boost`
- `Current runtime state same : No`

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# no power inline redundant
Router(config)#
*Jan 31 03:42:40.947: %PLATFORM_POWER-6-MODEMISMATCH: Inline power not in Boost mode
Router(config)#
Router(config)# exit
Router#
*Jan 31 03:36:13.111: %SYS-5-CONFIG_I: Configured from console by console
Router# show power
Main PSU :
Configured Mode : Boost
Current runtime state same : Yes
Total power available : 1450 Watts
POE Module :
Configured Mode : Boost
Current runtime state same : No
Total power available : 500 Watts
Router#
```

Available PoE Power

For the PoE feature to be available on the external PoE module, the total power from the power supplies must be 500 W or higher.

Note To ensure the PoE feature is functional on the external PoE module, verify the availability of PoE power on your router using the **show platform** and **show power** commands.

To determine there is enough PoE power for use by an external PoE service module, use the **show platform** and **show power** commands to calculate the available PoE power based on the wattage values of the main power supplies and PoE inverters.

Take the values of your main P0 and P1 power supplies to give the Total Power (for main power supplies.) Then take the values of your PoE1 and PoE2 power inverters to calculate the Total PoE Power.

The following table shows example modes of operation, which may be similar to your configuration.

The Total PoE Power value, in the final column of the table needs to be 500 W or higher for the PoE feature to be functional on a connected PoE service module.

Note Add power inverters to the router before inserting an external PoE module. Otherwise, even if the Total PoE Power is sufficient, the PoE power will not be used by the external PoE module and the module will need to be re-booted for the PoE feature to be functional.

Configuring a power mode of boost or redundant on the main power supplies, or PoE inverters, may affect the value for Total PoE Power.

The following table shows all power values in Watts. The wattage ratings of the main power supplies are shown in columns Main P0 and Main P1. The wattage ratings of the PoE inverters are shown in columns PoE0 and PoE1.

Table 28: Modes of Operation

Mode Example	Main P0	Main P1	Config Mode	Total Power (Main)	PoE0	PoE1	Config Mode	Total PoE Power
1	450	None	Redundant or Boost	450	None	500	Redundant or Boost	0 (None)
2	450	450	Boost	900	None	500	Redundant or Boost	0 (None)
3	450	450	Redundant	450	500	None	Redundant or Boost	0 (None)
4	1000	None	Redundant or Boost	1000	500	None	Redundant or Boost	500
5	1000	450	Redundant	450	500	500	Redundant or Boost	0 (None)
6	1000	450	Boost	1450	500	500	Boost	500
7	1000	1000	Redundant	1000	500	500	Boost	500
8	1000	1000	Boost	2000	500	500	Boost	1000

Note In the table above, for 500 W or higher Total PoE Power to be available, the "Total Power" (of the main power supplies) must be 1000 W or higher.

For 1000 W Total PoE Power (see Mode Example 8 above), there must be two 1000 W main power supplies (in `Boost` mode) and two PoE inverters (also in `Boost` mode).

Caution Care should be taken while removing the power supplies and power inverters (especially in `Boost` mode of operation). If the total power consumption is higher than can be supported by one power supply alone and in this condition a power supply is removed, the hardware can be damaged. This may then result in the system being unstable or unusable.

Similarly, in the case where there is only one PoE inverter providing PoE power to a service module, and in this condition the PoE inverter is removed, the hardware may be damaged, and may result in the system being unstable or unusable.

Managing PoE

The Power over Ethernet (PoE) feature allows you to manage power on the FPGE ports. By using PoE, you do not need to supply connected PoE-enabled devices with wall power. This eliminates the cost for additional electrical cabling that would otherwise be necessary for connected devices. The router supports PoE (802.3af) and PoE+ (802.3at). PoE provides up to 15.4 W of power, and PoE+ provides up to 30 W of power.

PoE Support for FPGE Ports

A PoE module supports PoE on the front panel gigabit ethernet ports (FPGE) such as gig0/0/0 and gig0/0/1. You can configure the PoE service module for the FPGE using the **power inline** command, which allows you to turn on or turn off the power to a connected device such as an IEEE phone or device. For more information, see Configuring PoE for FPGE Ports, on page 250.

Monitoring Your Power Supply

You can monitor the total available power budget on your router using the **show power inline [GigabitEthernet detail]** command in privileged EXEC mode.

This command allows you to check the availability of sufficient power for the powered device type before it is connected to the router.

Example—Inline power where there is no PoE module

In this example, there is no module present that supports PoE. Power is being supplied to an IP phone and a switch.

```
Router# show power inline
Available:31.0(w)   Used:30.3(w)   Remaining:0.7(w)

Interface Admin  Oper       Power   Device              Class Max
                            (Watts)
--------- ------ ---------- ------- ------------------- ----- ----
```

```
Gi0/0/0   auto   on         14.9    IP Phone 7971       3     30.0
Gi0/0/1   auto   on         15.4    WS-C2960CPD-8PT-L   4     30.0
Router#
```

In this example, the command includes the following information:

Available:31.0(w)—Available PoE power

Used:30.3(w)—PoE power used by all the router's ports

Oper—PoE power state of each connected powered device (on/off)

Power—PoE power used by each connected powered device

Class—PoE power classification

Example—Inline power for one PoE module

In this example, one module that supports PoE is present. Cisco IOS XE 3.10 and higher supports an external PoE module.

```
Router# show power inline
Available:31.0(w)  Used:30.3(w)  Remaining:0.7(w)

Interface Admin  Oper       Power   Device              Class Max
                            (Watts)
--------- ------ ---------- ------- ------------------- ----- ----
Gi0/0/0   auto   on         14.9    IP Phone 7971       3     30.0
Gi0/0/1   auto   on         15.4    WS-C2960CPD-8PT-L   4     30.0

Available:500.0(w)  Used:11.7(w)  Remaining:488.3(w)

Interface Admin  Oper       Power   Device              Class Max
                            (Watts)
--------- ------ ---------- ------- ------------------- ----- ----
Et2/0/0   auto   off        11.7    n/a                 n/a   750.0
Router#
```

Example—Inline power to connected IP phones

```
Router# show power inline
Available:31.0(w)  Used:30.8(w)  Remaining:0.2(w)

Interface Admin  Oper       Power   Device              Class Max
                            (Watts)
--------- ------ ---------- ------- ------------------- ----- ----
Gi0/0/0   auto   on         15.4    Ieee PD             4     30.0
Gi0/0/1   auto   on         15.4    Ieee PD             4     30.0
```

Example—Inline power to one Gigabit Ethernet port

```
Router# show power inline gigabitEthernet 0/0/0
Interface Admin  Oper       Power   Device              Class Max
                            (Watts)
--------- ------ ---------- ------- ------------------- ----- ----
Gi0/0/0   auto   on         15.4    Ieee PD             4     30.0
```

Example—Inline power to one Gigabit Ethernet port-detail

```
Router# show power inline gigabitEthernet 0/0/0 detail
Interface: Gi0/0/0
 Inline Power Mode: auto
 Operational status: on
 Device Detected: yes
 Device Type: Ieee PD
 IEEE Class: 4
 Discovery mechanism used/configured: Ieee
 Police: off

 Power Allocated
 Admin Value: 30.0
 Power drawn from the source: 15.4
 Power available to the device: 15.4

 Absent Counter: 0
 Over Current Counter: 0
 Short Current Counter: 0
 Invalid Signature Counter: 0
 Power Denied Counter: 0
```

Example—Inline power to an external PoE service module

In this example, after the output lines for Gi0/0/0, and Gi0/0/1, there are output lines for the external PoE service module. Cisco IOS XE 3.10 and higher supports an external PoE module. Et1/0/0 indicates the internal port (slot 1/0) for the first PoE service module. Et2/0/0 indicates the internal port (slot 2/0) in a second PoE service module.

Although both slots are capable of drawing 750 W of PoE power, in this device only 500 W of PoE power is available. Slot 2/0 (Et2/0/0) has been allocated 369.6 W of PoE power.

```
Router# show power inline
Available:31.0(w)  Used:15.4(w)  Remaining:15.6(w)
Interface Admin  Oper       Power   Device              Class Max
                            (Watts)
--------  ----   --------   -----   -----------------   ---   ---
Gi0/0/0   auto   on         15.4    Ieee PD             4     30.0
Gi0/0/1   auto   off        0.0     n/a                 n/a   30.0

Available:500.0(w)  Used:369.6(w)  Remaining:500.0(w)
Interface Admin  Oper       Power   Device              Class Max
                            (Watts)
--------  ----   --------   -----   -----------------   ---   ---
Et1/0/0   auto   off        0.0     n/a                 n/a   750.
Et2/0/0   auto   off        369.6   n/a                 n/a   750.
```

Enabling Cisco Discovery Protocol

Cisco Discovery Protocol (CDP) is enabled by default on the router.

Note CDP is not enabled by default on Cisco Aggregation Services Routers or on the Cisco CSR 1000v.

For more information on using CDP, see Cisco Discovery Protocol Configuration Guide, Cisco IOS XE Release 3S.

Configuring PoE for FPGE Ports

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cdp run**
4. **interface gigabitethernet** *slot/subslot/port*
5. **cdp enable**
6. **power inline** {**auto** { **auto** [**max** *milli-watts*] | **never**}
7. **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	**cdp run** **Example:** `Router(config)# cdp run`	Enables Cisco Discovery Protocol (CDP) on your router.
Step 4	**interface gigabitethernet** *slot/subslot/port* **Example:** `Router(config)# interface gigabitEthernet 0/0/0`	Allows to configure PoE on ports 0 and 1. • PoE can be configured on ports 0 and 1.
Step 5	**cdp enable** **Example:** `Router(config-if)#  cdp enable`	Enables CDP in the interface configuration mode.
Step 6	**power inline** {**auto** { **auto** [**max** *milli-watts*] | **never**} **Example:** `Router(config-if)# power inline auto`	Allows you to set the power inline options for FPGE ports. • **auto**—The **auto** keyword automatically detects the power inline devices and supplies power to such devices.

	Command or Action	Purpose
		• **max** *milli-watts*—The **max** keyword sets the maximum power allowed on the interface. • **never**—The **never** keyword disables the detection and ceases the application of inline power.
Step 7	**exit** **Example:** `Router(config-if)# exit`	Exits the interface configuration mode.

Verifying if PoE Is Enabled on FPGE Port

show platform: Example

show diag chassis eeprom: Example

You can verify whether the PoE is enabled on the FPGE port by looking at the external LED for this port. The external LED for the FPGE port is labelled as GE POE. The GE POE emits a green light when the internal PoE module is plugged in and functioning properly. The GE POE LED is yellow when the internal PoE is plugged in but not functioning properly. The GE POE LED is off when there are no PoE modules plugged in. For more information on LEDs, see the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

You can also detect PoE using the **show platform** and **show diag** commands.

For more information, see the following examples.

```
Router# show platform
Chassis type: ISR4451/K9

Chassis type: ISR4451/K9

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ISR4451/K9          ok                    3d11h
 0/0      ISR4451-X-4x1GE     ok                    3d11h
 0/2       NIM-4MFT-T1/E1    ok                    3d11h
 0/3      NIM-SSD             ok                    3d11h
1         ISR4451/K9          ok                    3d11h
 1/0      SM-X-1T3/E3         ok                    3d11h
2         ISR4451/K9          ok                    3d11h
 2/0      SM-ES3X-24-P        ok                    3d11h
R0        ISR4451/K9          ok, active            3d11h
F0        ISR4451/K9          ok, active            3d11h
P0        XXX-XXXX-XX         ok                    3d11h
P1        XXX-XXXX-XX         ok                    3d11h
P2        ACS-4451-FANTRAY    ok                    3d11h
POE0      PWR-POE-4451-X      ok                    3d11h
POE1      PWR-POE-4451-X      ok                    3d11h
GE-POE    800G2-POE-2         ok                    3d11h

Slot      CPLD Version        Firmware Version
--------- ------------------- ---------------------------------------
0         12090323            15.3(01r)S            [ciscouser-ISRRO...
```

```
1          12090323           15.3(01r)S          [ciscouser-ISRRO...
2          12090323           15.3(01r)S          [ciscouser-ISRRO...
R0         12090323           15.3(01r)S          [ciscouser-ISRRO...
F0         12090323           15.3(01r)S          [ciscouser-ISRRO...

Router# show diag chassis eeprom
MIDPLANE EEPROM data:

        Product Identifier (PID) : ISR-4451/K9
        Version Identifier (VID) : V01
        PCB Serial Number        : FOC16145VL8
        Hardware Revision        : 1.0
        Asset ID                 : P1C-R03-CP1.0-UMT-RVC
        CLEI Code                : TBD
Power/Fan Module P0 EEPROM data:

        Product Identifier (PID) : PWR-4450-AC
        Version Identifier (VID) : V01
        PCB Serial Number        : DCA1547X02U
        CLEI Code                : 0000000000
Power/Fan Module P1 EEPROM data is not initialized

Power/Fan Module P2 EEPROM data is not initialized

Internal PoE EEPROM data:

        Product Identifier (PID) : PWR-GE-POE-4400
        Version Identifier (VID) : V01
        PCB Serial Number        : FOC151849VD
        Hardware Revision        : 1.0
        CLEI Code                : 0000000000
```

Additional References

The following sections provide references related to the power efficiency management feature.

MIBs

MIBs	MIBs Link
CISCO-ENTITY-FRU-CONTROL-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use the Cisco MIB Locator at: http://www.cisco.com/go/mibs. Also see MIB Specifications Guide for the Cisco 4451-X Integrated Services Router.

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/cisco/web/support/index.html

CHAPTER 18

Factory Reset

This chapter describes Factory Reset feature and how it can be used to protect or restore a router to an earlier, fully functional state.

Feature Information for Factory Reset

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 https://cfnng.cisco.com/. An account on Cisco.com is not required.

Table 29: Feature Information for Factory Reset

Feature Name	Releases	Feature Information
Factory Reset	Cisco IOS XE Everest 16.6.1	This feature was introduced.
Secure Factory Reset with 3-pass or 7-pass	Cisco IOS XE Amsterdam 17.2.1	Added the **factory-reset all secure {3-pass | 7-pass}** command.
Option to retain RUM reports, SLR, and HSEC key using the **factory-reset keep-licensing-info**command	Cisco IOS XE Bengaluru 17.5.1	This feature was introduced.
Secure Factory Reset	Cisco IOS XE Dublin 17.12.1a	Added the **factory-reset all secure** command.

Information About Factory Reset

Factory reset is a process of clearing the current running and startup configuration information on a router, and resetting the router to an earlier, fully functional state.

From Cisco IOS XE Amsterdam XE 17.2 and later, you can use the **factory-reset all secure {3-pass | 7-pass}** command to clear the data in bootflash and ROMMON.

From Cisco IOS XE 17.12.1a, you can use the **factory-reset all secure** command to securely clear all the data in bootflash, hard disk, and ROMMON.

Note After the factory reset process is complete, the router reboots to ROMMON mode. If you have the zero-touch provisioning (ZTP) capability setup, after the router completes the factory reset procedure, the router reboots with ZTP configuration.

Table 30: Memory Components in ISR 4000 Series Routers

Component	Type	Sanitization
DRAM	Volatile	No sanitization required.
ROMMON	Non-Volatile	A factory reset using the **factory-reset all** command is the most common method used to erase customer data from the router's memory resources. The **factory-reset all secure** command (Cisco IOS XE 17.12.1a and later) can also be used to clear the data held in ROMMON in the same manner as the **factory-reset all** command.
Bootflash	Non-Volatile	A factory reset using the **factory-reset all** command is the most common method used to erase customer data from the router's memory resources. If additional flash memory is installed, the **factory-reset all** command will not erase the onboard flash memory. The **factory-reset all secure** command (Cisco IOS XE 17.12.1a and later) erases both the onboard and additional bootflash.

Component	Type	Sanitization
Harddisk	Non-Volatile	The **factory-reset all secure** command (Cisco IOS XE 17.12.1a and later) erases customer data from the hard disk.

Prerequisites for Performing Factory Reset

- Ensure that all the software images, configurations and personal data are backed up before performing factory reset.
- Ensure that there is uninterrupted power supply when factory reset is in progress.
- The factory reset process takes a backup of the boot image if the system is booted from an image stored locally (bootflash or hard disk). Ensure that you take a backup of the image before performing factory reset.
- The **factory-reset all secure** command clearly erases all files, including the boot image.

Restrictions for Performing a Factory Reset

- Any software patches that are installed on the router are not restored after the factory reset operation.
- If the factory reset command is issued through a Virtual Teletype (VTY) session, the session is not restored after the completion of the factory reset process.
- The **factory-reset all secure** command is supported only in the console, and not through a VTY session.

When to Perform Factory Reset

- Return Material Authorization (RMA): If a router is returned back to Cisco for RMA, it is important that all sensitive information is removed.
- Router is compromised: If the router data is compromised due to a malicious attack, the router must be reset to factory configuration and then reconfigured once again for further use.
- Repurposing: The router needs to be moved to a new topology or market from the existing site to a different site.

How to Perform a Factory Reset

Step 1 Log in to a Cisco 4000 Series ISR.

Important If the current boot image is a remote image or is stored in a USB or a NIM-SSD, ensure that you take a backup of the image before starting the factory reset process.

Step 2 This step is divided into three parts - a, b and c. If you need to retain the licensing information while performing the **factory-reset** command, follow step 2. a. If you do not need to retain licensing information and want all the data to be erased, perform step 2. b. If you do not need to retain licensing information and want all the data to be erased securely, perform step 2. c.

a) Execute **factory-reset keep-licensing-info** command to retain the licensing data.

The system displays the following message when you use the **factory-reset keep-licensing-info** command:

```
Router# factory-reset keep-licensing-info

The factory reset operation is irreversible for Keeping license usage. Are you sure? [confirm]
This operation may take 20 minutes or more. Please do not power cycle.

Dec 1 20:58:38.205: %PMAN-5-EXITACTION: R0/0: pvp: Process manager is exiting: process exit with
reload chassis code
/bootflash failed to mount
Dec 01 20:59:44.264: Factory reset operation completed.
Initializing Hardware ...

Current image running: Boot ROM1

Last reset cause: LocalSoft

ISR4331/K9 platform with 4194304 Kbytes of main memory
rommon 1
```

b) Execute the **factory-reset all** command to erase all data.

The system displays the following message when you use the **factory-reset all** command:

```
Router#factory-reset all

The factory reset operation is irreversible for all operations. Are you sure? [confirm]

This operation may take 20 minutes or more. Please do not power cycle.

*Jun 26 08:21:58.750: %SYS-5-RELOAD: Reload requested by Exec. Reload Reason: Factory Reset.
Jun 26 08:22:18.168: %PMAN-5-EXITACTION: R0/0: pvp: Process manager is exiting: process exit with
 reload chassis code
```

c) Execute one of the following commands: **factory-reset all secure** command, **factory-reset all secure 3-pass** command, or **factory-reset all secure 7-pass** command.

The system displays the following message when you use the **factory-reset all secure** command:

```
Router# factory-reset all secure
The factory reset operation is irreversible for securely reset all. Are you sure? [confirm]

This operation may take hours. Please do not power cycle.

*Feb 13 02:36:11.574: %SYS-5-RELOAD: Reload requested by Exec. Reload Reason: Factory Reset.Feb
13 02:36:19.379: %PMAN-5-EXITACTION: R0/0: pvp: Process manager is exiting: process exit with
reload chassis code

Enabling factory reset for this reload cycle

Feb 13 02:36:28.944: NIST 800 88r1 compliant factory reset starts.
Feb 13 02:36:29.027: #CISCO DATA SANITIZATION REPORT:# ISR4321/K9
Feb 13 02:36:29.112: start to purge non-volatile storage.
Executing Data Sanitization...
```

```
!!! Please, wait - mount bootflash !!!
!!! Please, wait - lsblk grep bootflash !!!
!!! Please, wait - umount bootflash !!!
bootflash:sdb, type:eusb-emmc found
!!! Please, wait - check spare flash info !!!
spare bootflash:sdc, type:eusb-emmc found
!!! Please, wait - lsblk -ln /dev/harddisk !!!
harddisk:sda, type:ssd found
eUSB-eMMC Data Sanitization started ...
!!! Please, wait - Reading eUSB-eMMC Info !!!
!!! Please, wait - Inquiring Unit Ready !!!
!!! Please, wait - Reading EXT_CSD !!!
!!! Please, wait - Reading EXT_CSD !!!
!!! Please, wait - Inquiring Unit Ready !!!
!!! Please, wait - Erasing(Secure-Trim1) /dev/sdb !!!
      Start Secure Trim1 (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Erasing(Secure-Trim1) /dev/sdb !!!
      Start Secure Trim1 (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Erasing(Secure-Trim1) /dev/sdb !!!
      Start Secure Trim1 (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Erasing(Secure-Trim1) /dev/sdb !!!
      Start Secure Trim1 (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Erasing(Secure-Trim2) /dev/sdb !!!
      Start Secure Trim2 (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Erasing(Secure-Trim2) /dev/sdb !!!
      Start Secure Trim2 (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Erasing(Secure-Trim2) /dev/sdb !!!
      Start Secure Trim2 (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Erasing(Secure-Trim2) /dev/sdb !!!
      Start Secure Trim2 (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Sanitizing /dev/sdb !!!
!!! Please, wait - Validating Erase for /dev/sdb !!!
eUSB-EMMC Data Sanitization completed ...
eUSB-eMMC Data Sanitization started ...
!!! Please, wait - Reading eUSB-eMMC Info !!!
!!! Please, wait - Inquiring Unit Ready !!!
!!! Please, wait - Reading EXT_CSD !!!
!!! Please, wait - Reading EXT_CSD !!!
!!! Please, wait - Inquiring Unit Ready !!!
!!! Please, wait - Erasing(Secure) /dev/sdc !!!
      Start Secure Erase (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Erasing(Secure) /dev/sdc !!!
      Start Secure Erase (CMD38) and Wait for a Maximum Of 120 Seconds for Good Completion
!!! Please, wait - Sanitizing /dev/sdc !!!
!!! Please, wait - Validating Erase for /dev/sdc !!!
eUSB-EMMC Data Sanitization completed ...
SSD Data Sanitization started ...
!!! Please, wait - Reading SSD Info !!!
!!! Please, wait - Reading SSD Info !!!
return code = 2
---
!!! Please, wait - Checking Sanitize Support-2 !!!
return code = 22
---
!!! Please, wait - Checking Sanitize Support-1 !!!
!!! Please, wait - Checking Enh Secure Support !!!
!!! Please, wait - Check SSD Frozen !!!
!!! Please, wait - Check SSD Frozen !!!

!!! Please, wait - Shredding !!!
SSD Data Sanitization completed ...
Data Sanitization Success! Exiting...
Feb 13 04:07:33.171: purge non-volatile storage done.
========================
```

```
#CISCO ISR4000 DATA SANITIZATION REPORT#
START : 13-02-2023, 02:36:32
  END : 13-02-2023, 04:07:30
-eUSB-eMMC-
MID : SMART(Hynix)
PNM : eUSB(JHBG4a2)
PRV : 2.11
Status : SUCCESS
NIST : PURGE
-eUSB-eMMC-
MID : CISCO(Hynix)
PNM : eMMC(JHAG2eeot)
PRV : 2.11
Status : SUCCESS
NIST : PURGE
-SSD-
MNM : SH9MST6D200GLE32C
SN : STP23340X9T
Status : SUCCESS
NIST : CLEAR
========================
Feb 13 04:07:33.746: start to check bootflash.
Feb 13 04:15:03.292: bootflash check done.
Feb 13 04:15:03.349: start to cleanup ROMMON variables.
Feb 13 04:15:07.629: ROMMON cleanup variables done.
Feb 13 04:15:07.699: start to cleanup ACT2/AIKIDO chip
Feb 13 04:15:10.879: ACT2/AIKIDO cleanup done.
Feb 13 04:15:10.953: report size:527
Feb 13 04:15:13.474: report save done.
Feb 13 04:15:13.525: Factory reset operation completed.
```

The system displays the following message when you use the **factory-reset all secure 3-pass** command:

```
Router# factory-reset all secure 3-pass

The factory reset operation is irreversible for securely reset all. Are you sure? [confirm]

This operation may take hours. Please do not power cycle.

*Jun 26 09:00:10.463: %SYS-5-RELOAD: Reload requested by Exec. Reload Reason: Factory Reset.Jun
26 09:00:19.461: %PMAN-5-EXITACTION: R0/0: pvp: Process manager is exiting: process exit with
reload chassis code
Enabling factory reset for this reload cycle

Jun 26 09:00:28.813: Factory reset secure operation. Write 0s. Please do not power cycle.
3812622336 bytes (3.8 GB, 3.6 GiB) copied, 132 s, 28.9 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 132.47 s, 28.9 MB/s
Jun 26 09:02:58.458: Factory reset secure operation. Write 1s. Please do not power cycle.
3821010944 bytes (3.8 GB, 3.6 GiB) copied, 145 s, 26.3 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 145.281 s, 26.3 MB/s
Jun 26 09:05:41.000: Factory reset secure operation. Write random. Please do not power cycle.
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 164 s, 23.3 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 164.079 s, 23.3 MB/s
Jun 26 09:08:42.913: Factory reset operation completed.
```

The system displays the following message when you use the **factory-reset all secure 7-pass** command:

```
Router# factory-reset all secure 7-pass

The factory reset operation is irreversible for securely reset all. Are you sure? [confirm]

This operation may take hours. Please do not power cycle.

*Jun 26 10:01:53.942: %SYS-5-RELOAD: Reload requested by Exec. Reload Reason: Factory Reset.Jun

Enabling factory reset for this reload cycle

Enabling
Jun 26 10:03:42.826: Factory reset secure operation. Write 0s. Please do not power cycle.
3816816640 bytes (3.8 GB, 3.6 GiB) copied, 137 s, 27.9 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 137.333 s, 27.9 MB/s
Jun 26 10:06:17.336: Factory reset secure operation. Write 1s. Please do not power cycle.
3804233728 bytes (3.8 GB, 3.5 GiB) copied, 142 s, 26.8 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 142.887 s, 26.8 MB/s
Jun 26 10:08:57.461: Factory reset secure operation. Write random. Please do not power cycle.
3816816640 bytes (3.8 GB, 3.6 GiB) copied, 163 s, 23.4 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 163.532 s, 23.4 MB/s
Jun 26 10:11:58.844: Factory reset secure operation. Write random. Please do not power cycle.
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 164 s, 23.3 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 164.145 s, 23.3 MB/s
Jun 26 10:15:00.804: Factory reset secure operation. Write 0s. Please do not power cycle.
3808428032 bytes (3.8 GB, 3.5 GiB) copied, 131 s, 29.1 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 131.586 s, 29.1 MB/s
Jun 26 10:17:29.774: Factory reset secure operation. Write 1s. Please do not power cycle.
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 145 s, 26.4 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 145.048 s, 26.4 MB/s
Jun 26 10:20:12.169: Factory reset secure operation. Write random. Please do not power cycle.
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 164 s, 23.3 MB/s
dd: error writing '/dev/bootflash': No space left on device
913+0 records in
912+0 records out
3825205248 bytes (3.8 GB, 3.6 GiB) copied, 164.111 s, 23.3 MB/s
Jun 26 10:23:14.166: Factory reset operation completed.
```

Step 3 Enter **confirm** to proceed with the factory reset.

Note

- If you want to quit the factory reset process, press the **Escape** key.
- The duration of the factory reset process depends on the storage size of the router. It can extend between 30 minutes and up to 3 hours on a high availability setup. If you want to quit the factory reset process, press the **Escape** key.

What Happens after a Factory Reset

After the factory reset is successfully completed, the router boots up. However, before the factory reset process started, if the configuration register was set to manually boot from ROMMON, the router stops at ROMMON.

After you configure Smart Licensing, execute the **#show license status** command, to check whether Smart Licensing is enabled for your instance.

Note If you had Specific License Reservation enabled before you performed the factory reset, use the same license and enter the same license key that you received from the smart agent.

CHAPTER 19

Configuring High Availability

The Cisco High Availability (HA) technology enable network-wide protection by providing quick recovery from disruptions that may occur in any part of a network. A network's hardware and software work together with Cisco High Availability technology, which besides enabling quick recovery from disruptions, ensures fault transparency to users and network applications.

The following sections describe how to configure Cisco High Availability features on your router:

About Cisco High Availability

The unique hardware and software architecture of your router is designed to maximize router uptime during any network event, and thereby provide maximum uptime and resilience within any network scenario.

This section covers some aspects of Cisco High Availability that may be used on the Cisco 4000 series routers:

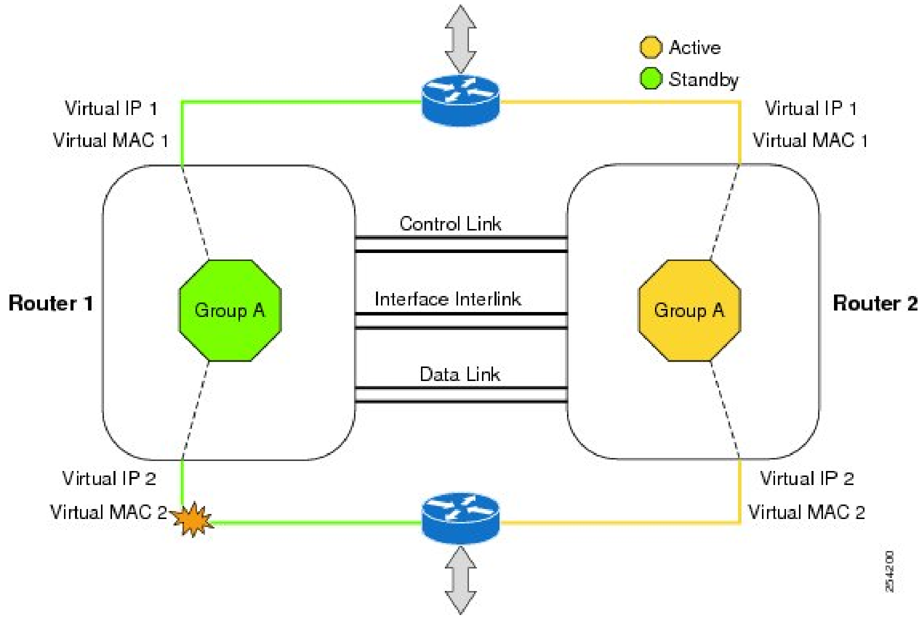
- Interchassis High Availability, on page 263
- Bidirectional Forwarding Detection, on page 264

Interchassis High Availability

The Interchassis High Availability feature is also known as the box-to-box redundancy feature. Interchassis High Availability enables the configuration of pairs of routers to act as backup for each other. This feature can be configured to determine the active router based on several failover conditions. When a failover occurs, the standby router seamlessly takes over and starts processing call signaling and performing media forwarding tasks.

Groups of redundant interfaces are known as redundancy groups. The following figure depicts the active-standby device scenario. It shows how the redundancy group is configured for a pair of routers that have a single outgoing interface.

Figure 2: Redundancy Group Configuration



The routers are joined by a configurable control link and data synchronization link. The control link is used to communicate the status of the routers. The data synchronization link is used to transfer stateful information to synchronize the stateful database for the calls and media flows. Each pair of redundant interfaces are configured with the same unique ID number, also known as the RII. For information on configuring Interchassis HA on your router, see Configuring Interchassis High Availability, on page 265.

IPsec Failover

The IPsec Failover feature increases the total uptime (or availability) of your IPsec network. Traditionally, the increased availability of your IPsec network is accomplished by employing a redundant (standby) router in addition to the original (active) router. When the active router becomes unavailable for a reason, the standby router takes over the processing of IKE and IPsec. IPsec failover falls into two categories: stateless failover and stateful failover.

On the router, only the stateless form of IPsec failover is supported. This stateless failover uses protocols such as the Hot Standby Router Protocol (HSRP) to provide primary to secondary cutover and also allows the active and standby VPN gateways to share a common virtual IP address.

Bidirectional Forwarding Detection

Bidirectional Forwarding Detection (BFD) is a detection protocol designed to provide fast-forwarding path-failure detection times for all media types, encapsulations, topologies, and routing protocols. In addition to fast-forwarding path-failure detection, BFD provides a consistent failure detection method for network administrators. Because a network administrator can use BFD to detect forwarding path failures at a uniform rate rather than variable rates for different routing protocol hello mechanisms, network profiling and planning is easier, and reconvergence time is consistent and predictable.

For more information on BFD, see the “Bidirectional Forwarding Detection” section in the IP Routing BFD Configuration Guide, Cisco IOS XE Release 3S.

Bidirectional Forwarding Detection Offload

The Bidirectional Forwarding Detection Offload feature allows the offload of BFD session management to the forwarding engine for improved failure detection times. BFD offload reduces the overall network convergence time by sending rapid failure detection packets (messages) to the routing protocols for recalculating the routing table. See Configuring BFD Offload, on page 266.

Configuring Cisco High Availability

Configuring Interchassis High Availability

Prerequisites

- The active device and the standby device must run on the identical version of the Cisco IOS XE software.
- The active device and the standby device must be connected through an L2 connection for the control path.
- The Embedded Service Processor (ESP) must be the same on both the active and standby devices. Route processors must also match and have a similar physical configuration.
- Either the Network Time Protocol (NTP) must be configured or the clock must be set identical on both devices to allow timestamps and call timers to match.
- Virtual router forwarding (VRF) must be defined in the same order on both active and standby routers for an accurate synchronization of data.
- The latency times must be minimal on all control and data links to prevent timeouts.
- Physically redundant links, such as Gigabit EtherChannel, must be used for the control and data paths.

Restrictions

- The failover time for a box-to-box application is higher for a non-box-to-box application.
- LAN and MESH scenarios are not supported.
- VRFs are not supported and cannot be configured under ZBFW High Availability data and control interfaces.
- The maximum number of virtual MACs (and VRFs) supported by the Front Panel Gigabit Ethernet (FPGE) interfaces depends on the platform. The supported Interfaces and Modules are listed in the Interfaces and Modules page. The Cisco 4400 Series ISRs FPGE support two reserved MACs and 24 filters which can be shared across all four FPGE interfaces. The Cisco 4300 Series ISRs FPGE support a maximum of 16 MACs with one reserved (BIA) and 15 filters. The NIM-1GE-CU-SFP,

NIM-2GE-CU-SFP, SM-X-6X1G, and SM-X-4X1G-1X10G modules, each port supports 1023 MAC filters. For information about the supported MAC filters for modules not listed, contact your Cisco representative.

> **Note** For information about limitations on sub-interfaces in HA configuration, see the section MAC Filter Distribution .

- When the configuration is replicated to the standby router, it is not committed to the startup configuration; it is in the running configuration. A user must run the **write memory** command to commit the changes that have been synchronized from the active router, on the standby router.

How to Configure Interchassis High Availability

For more information on configuring Interchassis High Availability on the router, see the IP Addressing: NAT Configuration Guide, Cisco IOS XE Release 3S.

Configuring Bidirectional Forwarding

For information on configuring BFD on your router, see the IP Routing BFD Configuration Guide.

For BFD commands, see the Cisco IOS IP Routing: Protocol-Independent Command Reference document.

Configuring BFD Offload

Restrictions

- Only BFD version 1 is supported.
- When configured, only offloaded BFD sessions are supported;, BFD session on RP are not supported.
- Only Asynchronous mode or no echo mode of BFD is supported.
- 511 asynchronous BFD sessions are supported.
- BFD hardware offload is supported for IPv4 sessions with non-echo mode only.
- BFD offload is supported only on port-channel interfaces.
- BFD offload is supported only for the Ethernet interface.
- BFD offload is not supported for IPv6 BFD sessions.
- BFD offload is not supported for BFD with TE/FRR.

How to Configure BFD Offload

BFD offload functionality is enabled by default. You can configure BFD hardware offload on the route processor. For more information, see Configuring BFD and the IP Routing BFD Configuration Guide.

Verifying Interchassis High Availability

Use the following **show** commands to verify the Interchassis High Availability.

Note Prerequisites and links to additional documentation configuring Interchassis High Availability are listed in Configuring Interchassis High Availability, on page 265.

- **show redundancy application group [group-id | all]**
- **show redundancy application transport {client | group [group-id]}**
- **show redundancy application control-interface group [group-id]**
- **show redundancy application faults group [group-id]**
- **show redundancy application protocol {protocol-id | group [group-id]}**
- **show redundancy application if-mgr group [group-id]**
- **show redundancy application data-interface group [group-id]**

The following example shows the redundancy application groups configured on the router:

```
Router# show redundancy application group
Group ID    Group Name                      State
--------    ----------                      -----
1           Generic-Redundancy-1            STANDBY
2           Generic-Redundancy2             ACTIVE
```

The following example shows the details of redundancy application group 1:

```
Router# show redundancy application group 1
Group ID:1
Group Name:Generic-Redundancy-1

Administrative State: No Shutdown
Aggregate operational state : Up
My Role: STANDBY
Peer Role: ACTIVE
Peer Presence: Yes
Peer Comm: Yes
Peer Progression Started: Yes

RF Domain: btob-one
RF state: STANDBY HOT
Peer RF state: ACTIVE
```

The following example shows the details of redundancy application group 2:

```
Router# show redundancy application group 2
Group ID:2
Group Name:Generic-Redundancy2

Administrative State: No Shutdown
Aggregate operational state : Up
My Role: ACTIVE
Peer Role: STANDBY
Peer Presence: Yes
Peer Comm: Yes
Peer Progression Started: Yes
```

```
RF Domain: btob-two
RF state: ACTIVE
Peer RF state: STANDBY HOT
```

The following example shows details of the redundancy application transport client:

```
Router# show redundancy application transport client
Client         Conn#  Priority   Interface   L3        L4
( 0)RF           0      1          CTRL       IPV4      SCTP

( 1)MCP_HA       1      1          DATA       IPV4      UDP_REL

( 4)AR           0      1          ASYM       IPV4      UDP

( 5)CF           0      1          DATA       IPV4      SCTP
```

The following example shows configuration details for the redundancy application transport group:

```
Router# show redundancy application transport group
Transport Information for RG (1)
Client = RF
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
0    0       10.1.1.1        59000    10.2.2.2         59000    CTRL    IPV4    SCTP
Client = MCP_HA
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
1    1       10.9.9.2        53000    10.9.9.1         53000    DATA    IPV4    UDP_REL
Client = AR
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
2    0       10.0.0.0        0        10.0.0.0         0        NONE_IN NONE_L3 NONE_L4
Client = CF
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
3    0       10.9.9.2        59001    10.9.9.1         59001    DATA    IPV4    SCTP
Transport Information for RG (2)
Client = RF
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
8    0       10.1.1.1        59004    10.1.1.2        59004    CTRL    IPV4    SCTP
Client = MCP_HA
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
9    1       10.9.9.2        53002    10.9.9.1         53002    DATA    IPV4    UDP_REL
Client = AR
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
10   0       10.0.0.0        0        10.0.0.0         0        NONE_IN NONE_L3 NONE_L4
Client = CF
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
11   0       10.9.9.2        59005    10.9.9.1         59005    DATA    IPV4    SCTP
```

The following example shows the configuration details of redundancy application transport group 1:

```
Router# show redundancy application transport group 1
Transport Information for RG (1)
Client = RF
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
0    0       10.1.1.1        59000    10.1.1.2        59000    CTRL    IPV4    SCTP
Client = MCP_HA
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
1    1       10.9.9.2        53000    10.9.9.1         53000    DATA    IPV4    UDP_REL
Client = AR
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
2    0       10.0.0.0        0        10.0.0.0         0        NONE_IN NONE_L3 NONE_L4
Client = CF
TI   conn_id my_ip           my_port  peer_ip          peer_por intf    L3      L4
3    0       10.9.9.2        59001    10.9.9.1         59001    DATA    IPV4    SCTP
```

The following example shows configuration details of redundancy application transport group 2:

```
Router# show redundancy application transport group 2
Transport Information for RG (2)
Client = RF
TI   conn_id my_ip          my_port  peer_ip        peer_por intf    L3      L4
8    0       10.1.1.1       59004    10.1.1.2       59004    CTRL    IPV4    SCTP
Client = MCP_HA
TI   conn_id my_ip          my_port  peer_ip        peer_por intf    L3      L4
9    1       10.9.9.2       53002    10.9.9.1       53002    DATA    IPV4    UDP_REL
Client = AR
TI   conn_id my_ip          my_port  peer_ip        peer_por intf    L3      L4
10   0       10.0.0.0       0        10.0.0.0       0        NONE_IN NONE_L3 NONE_L4
Client = CF
TI   conn_id my_ip          my_port  peer_ip        peer_por intf    L3      L4
11   0       10.9.9.2       59005    10.9.9.1       59005    DATA    IPV4    SCTP
```

The following example shows configuration details of the redundancy application control-interface group:

```
Router# show redundancy application control-interface group
The control interface for rg[1] is GigabitEthernet0/0/0
Interface is Control interface associated with the following protocols: 2 1
BFD Enabled
Interface Neighbors:
Peer: 10.1.1.2 Active RGs: 1 Standby RGs: 2 BFD handle: 0

The control interface for rg[2] is GigabitEthernet0/0/0
Interface is Control interface associated with the following protocols: 2 1
BFD Enabled
Interface Neighbors:
Peer: 10.1.1.2 Active RGs: 1 Standby RGs: 2 BFD handle: 0
```

The following example shows configuration details of the redundancy application control-interface group 1:

```
Router# show redundancy application control-interface group 1
The control interface for rg[1] is GigabitEthernet0/0/0
Interface is Control interface associated with the following protocols: 2 1
BFD Enabled
Interface Neighbors:
Peer: 10.1.1.2 Active RGs: 1 Standby RGs: 2 BFD handle: 0
```

The following example shows configuration details of the redundancy application control-interface group 2:

```
Router# show redundancy application control-interface group 2
The control interface for rg[2] is GigabitEthernet0/0/0
Interface is Control interface associated with the following protocols: 2 1
BFD Enabled
Interface Neighbors:
Peer: 10.1.1.2 Active RGs: 1 Standby RGs: 2 BFD handle: 0
```

The following example shows configuration details of the redundancy application faults group:

```
Router# show redundancy application faults group
Faults states Group 1 info:
Runtime priority: [50]
RG Faults RG State: Up.
Total # of switchovers due to faults: 0
Total # of down/up state changes due to faults: 2
Faults states Group 2 info:
Runtime priority: [135]
RG Faults RG State: Up.
Total # of switchovers due to faults: 0
Total # of down/up state changes due to faults: 2
```

The following example shows configuration details specific to redundancy application faults group 1:

```
Router# show redundancy application faults group 1
Faults states Group 1 info:
```

```
Runtime priority: [50]
RG Faults RG State: Up.
Total # of switchovers due to faults: 0
Total # of down/up state changes due to faults: 2
```

The following example shows configuration details specific to redundancy application faults group 2:

```
Router# show redundancy application faults group 2
Faults states Group 2 info:
Runtime priority: [135]
RG Faults RG State: Up.
Total # of switchovers due to faults: 0
Total # of down/up state changes due to faults: 2
```

The following example shows configuration details for the redundancy application protocol group:

```
Router# show redundancy application protocol group
RG Protocol RG 1
------------------
Role: Standby
Negotiation: Enabled
Priority: 50
Protocol state: Standby-hot
Ctrl Intf(s) state: Up
Active Peer: address 10.1.1.2, priority 150, intf Gi0/0/0
Standby Peer: Local
Log counters:
role change to active: 0
role change to standby: 1
disable events: rg down state 1, rg shut 0
ctrl intf events: up 2, down 1, admin_down 1
reload events: local request 0, peer request 0

RG Media Context for RG 1
--------------------------
Ctx State: Standby
Protocol ID: 1
Media type: Default
Control Interface: GigabitEthernet0/0/0
Current Hello timer: 3000
Configured Hello timer: 3000, Hold timer: 10000
Peer Hello timer: 3000, Peer Hold timer: 10000
Stats:
Pkts 117, Bytes 7254, HA Seq 0, Seq Number 117, Pkt Loss 0
Authentication not configured
Authentication Failure: 0
Reload Peer: TX 0, RX 0
Resign: TX 0, RX 0
Active Peer: Present. Hold Timer: 10000
Pkts 115, Bytes 3910, HA Seq 0, Seq Number 1453975, Pkt Loss 0



RG Protocol RG 2
------------------
Role: Active
Negotiation: Enabled
Priority: 135
Protocol state: Active
Ctrl Intf(s) state: Up
Active Peer: Local
Standby Peer: address 10.1.1.2, priority 130, intf Gi0/0/0
Log counters:
role change to active: 1
role change to standby: 1
```

```
disable events: rg down state 1, rg shut 0
ctrl intf events: up 2, down 1, admin_down 1
reload events: local request 0, peer request 0

RG Media Context for RG 2
--------------------------
Ctx State: Active
Protocol ID: 2
Media type: Default
Control Interface: GigabitEthernet0/0/0
Current Hello timer: 3000
Configured Hello timer: 3000, Hold timer: 10000
Peer Hello timer: 3000, Peer Hold timer: 10000
Stats:
Pkts 118, Bytes 7316, HA Seq 0, Seq Number 118, Pkt Loss 0
Authentication not configured
Authentication Failure: 0
Reload Peer: TX 0, RX 0
Resign: TX 0, RX 1
Standby Peer: Present. Hold Timer: 10000
Pkts 102, Bytes 3468, HA Seq 0, Seq Number 1453977, Pkt Loss 0
```

The following example shows configuration details for the redundancy application protocol group 1:

```
Router# show redundancy application protocol group 1
RG Protocol RG 1
------------------
Role: Standby
Negotiation: Enabled
Priority: 50
Protocol state: Standby-hot
Ctrl Intf(s) state: Up
Active Peer: address 10.1.1.2, priority 150, intf Gi0/0/0
Standby Peer: Local
Log counters:
role change to active: 0
role change to standby: 1
disable events: rg down state 1, rg shut 0
ctrl intf events: up 2, down 1, admin_down 1
reload events: local request 0, peer request 0

RG Media Context for RG 1
--------------------------
Ctx State: Standby
Protocol ID: 1
Media type: Default
Control Interface: GigabitEthernet0/0/0
Current Hello timer: 3000
Configured Hello timer: 3000, Hold timer: 10000
Peer Hello timer: 3000, Peer Hold timer: 10000
Stats:
Pkts 120, Bytes 7440, HA Seq 0, Seq Number 120, Pkt Loss 0
Authentication not configured
Authentication Failure: 0
Reload Peer: TX 0, RX 0
Resign: TX 0, RX 0
Active Peer: Present. Hold Timer: 10000
Pkts 118, Bytes 4012, HA Seq 0, Seq Number 1453978, Pkt Loss 0
```

The following example shows configuration details for the redundancy application protocol group 2:

```
Router# show redundancy application protocol group 2
RG Protocol RG 2
------------------
Role: Active
```

```
Negotiation: Enabled
Priority: 135
Protocol state: Active
Ctrl Intf(s) state: Up
Active Peer: Local
Standby Peer: address 10.1.1.2, priority 130, intf Gi0/0/0
Log counters:
role change to active: 1
role change to standby: 1
disable events: rg down state 1, rg shut 0
ctrl intf events: up 2, down 1, admin_down 1
reload events: local request 0, peer request 0

RG Media Context for RG 2
--------------------------
Ctx State: Active
Protocol ID: 2
Media type: Default
Control Interface: GigabitEthernet0/0/0
Current Hello timer: 3000
Configured Hello timer: 3000, Hold timer: 10000
Peer Hello timer: 3000, Peer Hold timer: 10000
Stats:
Pkts 123, Bytes 7626, HA Seq 0, Seq Number 123, Pkt Loss 0
Authentication not configured
Authentication Failure: 0
Reload Peer: TX 0, RX 0
Resign: TX 0, RX 1
Standby Peer: Present. Hold Timer: 10000
Pkts 107, Bytes 3638, HA Seq 0, Seq Number 1453982, Pkt Loss 0
```

The following example shows configuration details for the redundancy application protocol 1:

```
Router# show redundancy application protocol 1
Protocol id: 1, name: rg-protocol-1
BFD: ENABLE
Hello timer in msecs: 3000
Hold timer in msecs: 10000
OVLD-1#show redundancy application protocol 2
Protocol id: 2, name: rg-protocol-2
BFD: ENABLE
Hello timer in msecs: 3000
Hold timer in msecs: 10000
```

The following example shows configuration details for redundancy application interface manager group:

```
Router# show redundancy application if-mgr group
 RG ID: 1
 ==========

 interface      GigabitEthernet0/0/3.152
 ------------------------------------------
 VMAC           0007.b421.4e21
 VIP            10.1.1.255
 Shut           shut
 Decrement      10

 interface      GigabitEthernet0/0/2.152
 ------------------------------------------
 VMAC           0007.b421.5209
 VIP            10.1.2.255
 Shut           shut
 Decrement      10
```

```
 RG ID: 2
 ==========

 interface      GigabitEthernet0/0/3.166
 ---------------------------------------
 VMAC           0007.b422.14d6
 VIP            10.1.255.254
 Shut           no shut
 Decrement      10

 interface      GigabitEthernet0/0/2.166
 ---------------------------------------
 VMAC           0007.b422.0d06
 VIP            10.2.255.254
 Shut           no shut
 Decrement      10
```

The following examples shows configuration details for redundancy application interface manager group 1 and group 2:

```
Router# show redundancy application if-mgr group 1

 RG ID: 1
 ==========

 interface      GigabitEthernet0/0/3.152
 ---------------------------------------
 VMAC           0007.b421.4e21
 VIP            10.1.1.255
 Shut           shut
 Decrement      10

 interface      GigabitEthernet0/0/2.152
 ---------------------------------------
 VMAC           0007.b421.5209
 VIP            10.2.1.255
 Shut           shut
 Decrement      10

Router# show redundancy application if-mgr group 2
 RG ID: 2
 ==========

 interface      GigabitEthernet0/0/3.166
 ---------------------------------------
 VMAC           0007.b422.14d6
 VIP            10.1.255.254
 Shut           no shut
 Decrement      10

 interface      GigabitEthernet0/0/2.166
 ---------------------------------------
 VMAC           0007.b422.0d06
 VIP            10.2.255.254
 Shut           no shut
 Decrement      10
```

The following example shows configuration details for redundancy application data-interface group:

```
Router# show redundancy application data-interface group
The data interface for rg[1] is GigabitEthernet0/0/1
The data interface for rg[2] is GigabitEthernet0/0/1
```

The following examples show configuration details specific to redundancy application data-interface group 1 and group 2:

```
Router# show redundancy application data-interface group 1
The data interface for rg[1] is GigabitEthernet0/0/1

Router # show redundancy application data-interface group 2
The data interface for rg[2] is GigabitEthernet0/0/1
```

Verifying BFD Offload

Use the following commands to verify and monitor BFD offload feature on your router.

Note Configuration of BFD Offload is described in Configuring Bidirectional Forwarding, on page 266.

- **show bfd neighbors [details]**
- **debug bfd [packet | event]**
- **debug bfd event**

The **show bfd neighbors** command displays the BFD adjacency database:

```
Router# show bfd neighbor

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
192.0.2.10                            362/1277       Up        Up        Gi0/0/1.2
192.0.2.11                            445/1278       Up        Up        Gi0/0/1.3
192.0.2.12                           1093/961        Up        Up        Gi0/0/1.4
192.0.2.13                           1244/946        Up        Up        Gi0/0/1.5
192.0.2.14                           1094/937        Up        Up        Gi0/0/1.6
192.0.2.15                           1097/1260       Up        Up        Gi0/0/1.7
192.0.2.16                           1098/929        Up        Up        Gi0/0/1.8
192.0.2.17                           1111/928        Up        Up        Gi0/0/1.9
192.0.2.18                           1100/1254       Up        Up        Gi0/0/1.10
```

The **debug bfd neighbor detail** command displays the debugging information related to BFD packets:

```
Router# show bfd neighbor detail

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
192.0.2.10                            362/1277       Up        Up        Gi0/0/1.2
Session state is UP and not using echo function.
Session Host: Hardware
OurAddr: 192.0.2.11
Handle: 33
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 50(0)
Rx Count: 3465, Rx Interval (ms) min/max/avg: 42/51/46
Tx Count: 3466, Tx Interval (ms) min/max/avg: 39/52/46
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: CEF EIGRP
Uptime: 00:02:50
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 1
             Multiplier: 3               - Length: 24
```

```
            My Discr.: 1277             - Your Discr.: 362
            Min tx interval: 50000      - Min rx interval: 50000
            Min Echo interval: 0
```

The **show bfd summary** command displays the BFD summary:

```
Router# show bfd summary

                    Session          Up          Down

Total                   400         400             0
```

The **show bfd drops** command displays the number of packets dropped in BFD:

```
Router# show bfd drops
BFD Drop Statistics
                      IPV4     IPV6    IPV4-M    IPV6-M    MPLS_PW   MPLS_TP_LSP
Invalid TTL            0         0       0         0           0       0
BFD Not Configured     0         0       0         0           0       0
No BFD Adjacency       33        0       0         0           0       0
Invalid Header Bits    0         0       0         0           0       0
Invalid Discriminator  1         0       0         0           0       0
Session AdminDown      94        0       0         0           0       0
Authen invalid BFD ver 0         0       0         0           0       0
Authen invalid len     0         0       0         0           0       0
Authen invalid seq     0         0       0         0           0       0
Authen failed          0         0       0         0           0       0
```

The **debug bfd packet** command displays debugging information about BFD control packets.

```
Router# debug bfd packet
*Nov 12 23:08:27.982: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1941/0 diag:0(No Diagnostic)
 Down C cnt:4 ttl:254 (0)
*Nov 12 23:08:27.982: BFD-DEBUG Packet: Tx IP:192.0.2.22 ld/rd:983/1941 diag:3(Neighbor
Signaled Session Down) Init  C cnt:44 (0)
*Nov 12 23:08:28.007: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1941/983 diag:0(No Diagnostic)
 Up PC cnt:4 ttl:254 (0)
*Nov 12 23:08:28.007: BFD-DEBUG Packet: Tx IP:192.0.2.22 ld/rd:983/1941 diag:0(No Diagnostic)
 Up F C cnt:0 (0)
*Nov 12 23:08:28.311: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1941/983 diag:0(No Diagnostic)
 Up FC cnt:0 ttl:254 (0)
*Nov 12 23:08:28.311: BFD-DEBUG Packet: Tx IP:192.0.2.22 ld/rd:983/1941 diag:0(No Diagnostic)
 Up  C cnt:0 (0)
*Nov 12 23:08:28.311: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1907/0 diag:0(No Diagnostic)
 Down C cnt:3 ttl:254 (0)
*Nov 12 23:08:28.311: BFD-DEBUG Packet: Tx IP:192.0.2.22 ld/rd:993/1907 diag:3(Neighbor
Signaled Session Down) Init  C cnt:43 (0)
*Nov 12 23:08:28.311: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1941/983 diag:0(No Diagnostic)
 Up C cnt:0 ttl:254 (0)
*Nov 12 23:08:28.626: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1907/993 diag:0(No Diagnostic)
 Up PC cnt:3 ttl:254 (0)
*Nov 12 23:08:28.626: BFD-DEBUG Packet: Tx IP:192.0.2.22 ld/rd:993/1907 diag:0(No Diagnostic)
 Up F C cnt:0 (0)
*Nov 12 23:08:28.645: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1907/993 diag:0(No Diagnostic)
 Up C cnt:0 ttl:254 (0)
*Nov 12 23:08:28.700: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1907/993 diag:0(No Diagnostic)
 Up FC cnt:0 ttl:254 (0)
*Nov 12 23:08:28.700: BFD-DEBUG Packet: Tx IP:192.0.2.22 ld/rd:993/1907 diag:0(No Diagnostic)
 Up  C cnt:0 (0)
*Nov 12 23:08:28.993: BFD-DEBUG Packet: Rx IP:192.0.2.22 ld/rd:1907/993 diag:0(No Diagnostic)
 Up C cnt:0 ttl:254 (0)
```

The **debug bfd event** displays debugging information about BFD state transitions:

```
Router# deb bfd event
```

```
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(EIGRP) IP:192.0.2.153, ld:1401,
handle:77, event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.153, ld:1401, handle:77,
 event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(EIGRP) IP:192.0.2.153, ld:1400,
handle:39, event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.153, ld:1400, handle:39,
 event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(EIGRP) IP:192.0.2.153, ld:1399,
handle:25, event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.153, ld:1399, handle:25,
 event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(EIGRP) IP:192.0.2.153, ld:1403,
handle:173, event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.153, ld:1403, handle:173,
 event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(EIGRP) IP:192.0.2.153, ld:1402,
handle:95, event:DOWN adminDown, (0)
*Nov 12 23:11:29.503: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.153, ld:1402, handle:95,
 event:DOWN adminDown, (0)
*Nov 12 23:11:30.639: BFD-HW-API: Handle 1404: Timers: Tx timer 1000000 Detect timer 0
*Nov 12 23:11:30.639: BFD-HW-API: Handle 1404: Flags: Poll 0 Final 0
*Nov 12 23:11:30.639: BFD-HW-API: Handle 1404: Buffer: 0x23480318 0x0000057C 0x00000000
0x000F4240 0x000F4240 0x00000000 size 24
*Nov 12 23:11:30.641: BFD-HW-API: Handle 1405: Timers: Tx timer 1000000 Detect timer 0
*Nov 12 23:11:30.641: BFD-HW-API: Handle 1405: Flags: Poll 0 Final 0
*Nov 12 23:11:30.641: BFD-HW-API: Handle 1405: Buffer: 0x23480318 0x0000057D 0x00000000
0x000F4240 0x000F4240 0x00000000 size 24
*Nov 12 23:11:30.649: BFD-DEBUG Packet: Rx IP:192.0.2.33 ld/rd:1601/1404
diag:7(Administratively Down) AdminDown C cnt:0 ttl:254 (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: V1 FSM ld:1404 handle:207 event:RX ADMINDOWN state:UP
 (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: resetting timestamps ld:1404 handle:207 (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.33, ld:1404, handle:207,
 event:DOWN adminDown, (0)
*Nov 12 23:11:30.650: BFD-DEBUG Packet: Tx IP:192.0.2.33 ld/rd:1404/0 diag:3(Neighbor
Signaled Session Down) Down C cnt:0 (0)
*Nov 12 23:11:30.650: BFD-DEBUG Packet: Rx IP:192.0.2.85 ld/rd:1620/1405
diag:7(Administratively Down) AdminDown C cnt:0 ttl:254 (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: V1 FSM ld:1405 handle:209 event:RX ADMINDOWN state:UP
 (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: resetting timestamps ld:1405 handle:209 (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.85, ld:1405, handle:209,
 event:DOWN adminDown, (0)
*Nov 12 23:11:30.650: BFD-DEBUG Packet: Tx IP:192.10.85.1 ld/rd:1405/0 diag:3(Neighbor
Signaled Session Down) Down C cnt:0 (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: notify client(EIGRP) IP:192.0.2.33, ld:1404,
handle:207, event:DOWN adminDown, (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.33, ld:1404, handle:207,
 event:DOWN adminDown, (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: notify client(EIGRP) IP:192.0.2.85, ld:1405,
handle:209, event:DOWN adminDown, (0)
*Nov 12 23:11:30.650: BFD-DEBUG Event: notify client(CEF) IP:192.0.2.85, ld:1405, handle:209,
 event:DOWN adminDown, (0)
*Nov 12 23:11:31.035: %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 192.0.2.191
```

Additional References

The following documents provide information related to the BFD feature.

Related Topic	Document Title
Configuring Stateful Interchassis Configuration.	*Security Configuration Guide: Zone-Based Policy Firewall, Cisco IOS XE Release 3S* at: http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_data_zbf/configuration/xe-3s/sec-data-zbf-xe-book.html.
IP Routing Protocol-Independent Commands.	*Cisco IOS IP Routing: Protocol-Independent Command Reference* at: http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_pi/command/iri-cr-book.html.

CHAPTER 20

Secure Sockets Layer Virtual Private Network (SSL VPN)

The Secure Sockets Layer Virtual Private Network (SSL VPN) feature provides support in the Cisco IOS software for remote user access to enterprise networks from anywhere on the internet. Remote access is provided through a Secure Socket Layer-enabled (SSL-enabled) SSL VPN gateway. The SSL VPN gateway allows remote users to establish a secure VPN tunnel. The SSL VPN feature provides a comprehensive solution that allows easy access to a broad range of web resources and web-enabled applications using original HTTP over SSL (HTTPS) browser support through the full-tunnel client support.

Prerequisites for SSL VPN

To securely access resources on a private network behind an SSL VPN gateway, the remote user of an SSL VPN service must have the following:

- An account (login name and password).
- Support for full tunnel mode using Cisco AnyConnect client.
- Administrative privileges to install Cisco AnyConnect client.

Restrictions for SSL VPN

- ACLs do not support DENY statements.
- Using Cisco AnyConnect VPN, if you create tunnels at a high bring-up rate, a failure might occur. When creating a large number of VPN SSL sessions, for example, 1000, use a bring-up rate of 15 TPS or lower. If you use a higher TPS rate, a failure might occur.

- SSLVPN Peer Detection (PD) is supported only with AnyConnect client Version 3.x and later.

Information About SSL VPN

SSL VPN Overview

Cisco IOS XE SSL VPN is a router-based solution offering SSL VPN remote-access connectivity integrated with industry-leading security and routing features on a converged data, voice, and wireless platform. The security is transparent to end users and is easy to administer. With Cisco IOS XE SSL VPN, end users gain access securely from home or any internet-enabled location such as wireless hotspots. Cisco IOS XE SSL VPN also enables companies to extend corporate network access to offshore partners and consultants, keeping corporate data protected all the while. Cisco IOS XE SSL VPN, in conjunction with the dynamically downloaded Cisco AnyConnect VPN client, provides remote users with full network access to virtually any corporate application.

SSL VPN delivers the following three modes of SSL VPN access, of which only tunnel mode is supported in Cisco IOS XE software:

- Clientless: Clientless mode provides secure access to private web resources and to web content. This mode is useful for accessing most content that you would expect to access in a web browser, such as internet access, databases, and online tools that use a web interface.
- Thin Client (port-forwarding Java applet): Thin client mode extends the capability of the cryptographic functions of the web browser to enable remote access to TCP-based applications such as Post Office Protocol version 3 (POP3), Simple Mail Transfer Protocol (SMTP), Internet Message Access protocol (IMAP), Telnet, and Secure Shell (SSH).
- Full-Tunnel Mode: Full-tunnel client mode offers extensive application support through its dynamically downloaded Cisco AnyConnect VPN client (next-generation SSL VPN client) for SSL VPN. Full-tunnel client mode delivers a lightweight, centrally configured and easy-to-support SSL VPN tunneling client that provides network layer access to virtually any application.

Note SSL VPN will not work if **ip http secure-server** is enabled.

This feature is supported on the following platforms:

Platform	Supported Cisco IOS XE Release
Cisco Cloud Services Router 1000V Series	Cisco IOS XE Release 16.9
Cisco Catalyst 8000V	Cisco IOS XE Bengaluru 17.4.1
Cisco 4461 Integrated Services Router Cisco 4451 Integrated Services Router Cisco 4431 Integrated Services Router	Cisco IOS XE Cupertino 17.7.1a

Remote Access Modes

In a typical clientless remote access scenario, remote users establish an SSL tunnel to move data to and from the internal networks at the application layer, for example, web and email. In tunnel mode, remote users use an SSL tunnel to move data at the network (IP) layer. Therefore, tunnel mode supports most IP-based applications. Tunnel mode supports many popular corporate applications, for example, Microsoft Outlook, Microsoft Exchange, Lotus Notes E-mail, and Telnet.

SSL VPN support that is provided by full-tunnel mode is as follows:

- Works like clientless IPsec VPN
- Tunnel client loaded through Java or ActiveX
- Application agnostic; supports all IP-based applications
- Scalable
- Local administrative permissions required for installation

Full-tunnel client mode offers extensive application support through its dynamically downloaded Cisco AnyConnect VPN client (next-generation SSL VPN client) for SSL VPN. Full-tunnel client mode delivers a lightweight, centrally configured, and easy-to-support SSL VPN tunneling client that provides network layer access to virtually any application. The advantage of SSL VPN comes from its accessibility from almost any internet-connected system without needing to install additional desktop software. Cisco SSL AnyConnect VPN allows remote users to access enterprise networks on the internet through an SSL VPN gateway. During the establishment of the SSL VPN with the gateway, the Cisco AnyConnect VPN client is downloaded and installed on the remote user equipment (laptop, mobile, PDA, and so on. The tunnel connection is established when a remote user logs into the SSL VPN gateway. The tunnel connection is determined by the group policy configuration. By default, the Cisco AnyConnect VPN client is removed from the client PC after the connection is closed. However, you have the option to keep the Cisco AnyConnect VPN client installed on the client equipment.

Cisco SSL AnyConnect VPN easily accesses the services within the company's network and simplifies the VPN configuration on the SSL VPN gateway, thereby reducing the overhead for system administrators.

SSL VPN CLI Constructs

SSL Proposal

SSL proposal specifies the cipher suites that are supported. Each cipher suite defines a key exchange algorithm, a bulk encryption algorithm, and a MAC algorithm. One of the cipher suites that is configured would be chosen from the client's proposal during SSL negotiation. If the intersection between a client's proposed suites and configured suites is a null set, the negotiation terminates. Ciphers are currently selected based on the client's priority.

The SSL proposal is used in SSL handshake protocol for negotiating encryption and decryption. The default SSL proposal is used with SSL policy in the absence of any user-defined proposal. The default proposal has ciphers in the order shown here:

```
protection rsa-aes256-sha1 rsa-aes128-sha1 rsa-3des-ede-sha1 rsa-3des-ede-sha1
```

SSL Policy

SSL policy defines the cipher suites to be supported and the trust point to be used during SSL negotiation. SSL policy is a container of all the parameters used in the SSL negotiation. The policy selection is done by matching the session parameters against the parameters configured under the policy. There is no default policy. Every policy is associated with a proposal and a trustpoint.

SSL Profile

The SSL VPN profile defines authentication and accounting lists. A profile selection depends on policy and URL values. Profile may, optionally, be associated with a default authorization policy.

The following rules apply:

- The policy and URL must be unique for an SSL VPN profile.
- At least one authorization method must be specified to bring up the session.
- The three authorization types, namely user, group and cached can coexist.
- There is no default authorization.
- The order of precedence for authorization is user authorization, cache authorization, and group authorization. If group authorization override is configured, the order of precedence is group authorization, user authorization, and cache authorization.

SSL Authorization Policy

The SSL authorization policy is a container of authorization parameters that are pushed to a remote client and are applied either locally on the virtual-access interface, or globally on the device. The authorization policy is referred from the SSL VPN profile.

SSL VPN MIB

The SSL VPN MIB represents the Cisco implementation-specific attributes of a Cisco entity that implements SSL VPN. The MIB provides operational information in Cisco’s SSL VPN implementation by managing the SSL VPN, trap control, and notification groups. For example, the SSL VPN MIB provides the number of active SSL tunnels on the device.

How to Configure SSL VPN

The following sections provide information about the various tasks involved in configuring SSL VPN.

Configuring an SSL Proposal

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ssl proposal** *proposal-name*

4. **protection**
5. **end**
6. **show crypto ssl proposal** [*proposal name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**crypto ssl proposal** *proposal-name* **Example:** `Device(config)# crypto ssl proposal proposal1`	Defines an SSL proposal name, and enters crypto SSL proposal configuration mode.
Step 4	**protection** **Example:** `Device(config-crypto-ssl-proposal)# protection rsa-3des-ede-sha1 rsa-aes128-sha1`	Specifies one or more cipher suites that are as follows: • rsa-3des-ede-sha1 • rsa-aes128-sha1 • rsa-aes256-sha1 • rsa-rc4128-md5
Step 5	**end** **Example:** `Device(config-crypto-ssl-proposal)# end`	Exits SSL proposal configuration mode and returns to privileged EXEC mode.
Step 6	**show crypto ssl proposal** [*proposal name*] **Example:** `Device# show crypto ssl proposal`	(Optional) Displays the SSL proposal.

Configuring an SSL Policy

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ssl policy** *policy-name*
4. **ip address local** *ip-address* [**vrf** *vrf-name*] [**port** *port-number*] [**standby** *redundancy-name*]
5. **ip interface local** *interface-name* [**vrf** *vrf-name*] [**port** *port-number*] [**standby** *redundancy-name*]
6. **pki trustpoint** *trustpoint-name* **sign**

7. **ssl proposal** *proposal-name*
8. **no shut**
9. **end**
10. **show crypto ssl policy** [*policy-name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**crypto ssl policy** *policy-name* **Example:** `Device(config)# crypto ssl policy policy1`	Defines an SSL policy name and enters SSL policy configuration mode.
Step 4	**ip address local** *ip-address* [**vrf** *vrf-name*] [**port** *port-number*] [**standby** *redundancy-name*] **Example:** `Device(config-crypto-ssl-policy)# ip address local 10.0.0.1 port 446`	Specifies the local IP address to start the TCP listener. **Note** Running this command or the **ip interface local** command is mandatory.
Step 5	**ip interface local** *interface-name* [**vrf** *vrf-name*] [**port** *port-number*] [**standby** *redundancy-name*] **Example:** `Device(config-crypto-ssl-policy)# ip interface local FastEthernet redundancy1`	Specifies the local interface to start the TCP listener. **Note** Running this command or the **ip address local** command is mandatory.
Step 6	**pki trustpoint** *trustpoint-name* **sign** **Example:** `Device(config-crypto-ssl-policy)# pki trustpoint tp1 sign`	(Optional) Specifies the trustpoint to be used to send the server certificate during an SSL handshake. **Note** If this command is not specified, a default self-signed trustpoint is used. If there is no default self-signed trustpoint, the system creates a default self-signed certificate.
Step 7	**ssl proposal** *proposal-name* **Example:** `Device(config-crypto-ssl-policy)# ssl proposal pr1`	(Optional) Specifies the cipher suites to be selected during an SSL handshake. **Note** If a proposal is not specified, the default proposal is used.
Step 8	**no shut** **Example:** `Device(config-crypto-ssl-policy)# no shut`	Starts the TCP listener based on the configuration.

	Command or Action	Purpose
Step 9	**end** **Example:** `Device(config-crypto-ssl-policy)# end`	Exits SSL policy configuration mode and returns to privileged EXEC mode.
Step 10	**show crypto ssl policy** [*policy-name*] **Example:** `Device# show crypto ssl policy`	(Optional) Displays the SSL policies.

Configuring an SSL Profile

Before you begin

For details of AAA configuration, see the *Authentication Authorization and Accounting Configuration Guide*.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ssl profile** *profile-name*
4. **aaa accounting user-pass list** *list-name*
5. **aaa authentication user-pass list** *list-name*
6. **aaa authorization group** [**override**] **user-pass list** *aaa-listname aaa-username*
7. **aaa authorization user user-pass** {**cached** | **list** *aaa-listname aaa-username*}
8. **match policy** *policy-name*
9. **match url** *url-name*
10. **no shut**
11. **end**
12. **show crypto ssl profile** [*profile-name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**crypto ssl profile** *profile-name* **Example:** `Device(config)# crypto ssl profile profile1`	Defines an SSL profile and enters SSL profile configuration mode.

	Command or Action	Purpose
Step 4	**aaa accounting user-pass list** *list-name* **Example:** `Device(config-crypto-ssl-profile)# aaa accounting user-pass list list1`	Specifies authentication, authorization, and accounting (AAA) method list.
Step 5	**aaa authentication user-pass list** *list-name* **Example:** `Device(config-crypto-ssl-profile)# aaa authentication user-pass list list2`	Specifies the AAA method list.
Step 6	**aaa authorization group** [**override**] **user-pass list** *aaa-listname aaa-username* **Example:** `Device(config-crypto-ssl-profile)# aaa authorization group override user-pass list list1 user1`	Specifies the AAA method list and username for group authorization. • **group**: Specifies group authorization. • **override**: (Optional) Specifies that attributes from group authorization should take precedence while merging attributes. By default, user attributes take precedence. • **user-pass**: Specifies the user password-based authorization. • *aaa-listname*: AAA method list name. • *aaa-username*: Username that must be used in the AAA request. Refers to the SSL authorization policy name defined on the device.
Step 7	**aaa authorization user user-pass** {**cached** | **list** *aaa-listname aaa-username*} **Example:** `Device(config-crypto-ssl-profile)# aaa authorization user user-pass list list1 user1`	Specifies the AAA method list and username for user authorization. • **user**—Specifies user authorization. • **user-pass**— Specifies the user password-based authorization. • **cached**—Specifies that the attributes received during EAP authentication or obtained from the AAA preshared key must be cached. • *aaa-listname*—AAA method list name. • *aaa-username*—Username that must be used in the AAA authorization request.
Step 8	**match policy** *policy-name* **Example:** `Device(config-crypto-ssl-profile)# match policy policy1`	Uses match statements to select an SSL profile for a peer based on the SSL policy name.

	Command or Action	Purpose
Step 9	**match url** *url-name* **Example:** `Device(config-crypto-ssl-profile)# match url www.abc.com`	Uses match statements to select an SSL profile for a peer based on the URL.
Step 10	**no shut** **Example:** `Device(config-crypto-ssl-profile)# no shut`	Specifies that profile cannot be shut until the policy specified in the **match policy** command is in use.
Step 11	**end** **Example:** `Device(config-crypto-ssl-profile)# end`	Exits SSL profile configuration mode and returns to privileged EXEC mode.
Step 12	**show crypto ssl profile** [*profile-name*] **Example:** `Device# show crypto ssl profile`	(Optional) Displays the SSL profile.

Configuring an SSL Authorization Policy

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ssl authorization policy** *policy-name*
4. **banner** *banner-text*
5. **client profile** *profile-name*
6. **def-domain** *domain-name*
7. Run one of the following commands:
   - **dns** *primary-server* [*secondary-server*]
   - Or
   - **ipv6 dns** *primary-server* [*secondary-server*]
8. **dpd-interval** {**client** | **server**} *interval*
9. **homepage** *homepage-text*
10. **include-local-lan**
11. **ipv6 prefix** *prefix*
12. **keepalive** *seconds*
13. **module** *module-name*
14. **msie-proxy exception** *exception-name*
15. **msie-proxy option** {**auto** | **bypass** | **none**}
16. **msie-proxy server** {*ip-address* | *dns-name*}
17. **mtu** *bytes*
18. **netmask** *mask*

19. Run one of the following commands:
    - **pool** *name*
    - Or
    - **ipv6 pool** *name*
20. **rekey time** *seconds*
21. Run one of the following commands:
    - **route set access-list** *acl-name*
    - Or
    - **ipv6 route set access-list** *access-list-name*
22. **smartcard-removal-disconnect**
23. **split-dns** *string*
24. **timeout** {**disconnect** *seconds* | **idle** *seconds* | **session** *seconds*}
25. **wins** *primary-server* [*secondary-server*]
26. **end**
27. **show crypto ssl authorization policy** [*policy-name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**crypto ssl authorization policy** *policy-name* **Example:** `Device(config)# crypto ssl authorization policy policy1`	Specifies the SSL authorization policy and enters SSL authorization policy configuration mode.
Step 4	**banner** *banner-text* **Example:** `Device(config-crypto-ssl-auth-policy)# banner This is SSL VPN tunnel. NOTE: DO NOT dial emergency response numbers (e.g. 911,112) from software telephony clients. Your exact location and the appropriate emergency response agency may not be easily identified.`	Specifies the banner. The banner is displayed after the successful setup of the tunnel.
Step 5	**client profile** *profile-name* **Example:** `Device(config-crypto-ssl-auth-policy)# client profile Employee`	Specifies the AnyConnect client profile. The profile must already be specified using the **crypto vpn anyconnect profile** command. See section Example: Specifying the AnyConnect Image and Profile, on page 296 for sample configuration of the AnyConnect image and profile.

	Command or Action	Purpose
		For details of AnyConnect configuration, see the Cisco AnyConnect Secure Mobility Client Administrator Guide.
Step 6	**def-domain** *domain-name* **Example:** `Device(config-crypto-ssl-auth-policy)# def-domain example.com`	Specifies the default domain. This parameter specifies the default domain that the client can use.

Step 7	Run one of the following commands:
	• **dns** *primary-server* [*secondary-server*]
	• Or
	• **ipv6 dns** *primary-server* [*secondary-server*]
	Example:
	`Device(config-crypto-ssl-auth-policy)# dns 198.51.100.1 198.51.100.100`
	Example:
	`Device(config-crypto-ssl-auth-policy)# ipv6 dns 2001:DB8:1::1 2001:DB8:2::2`
		Specifies an IPv4-based or IPv6-based address for the primary and secondary Domain Name Service (DNS) servers.
		• *primary-server*: IP address of the primary DNS server.
		• *secondary-server*: (Optional) IP address of the secondary DNS server.

Step 8	**dpd-interval** {**client** | **server**} *interval* **Example:** `Device(config-crypto-ssl-auth-policy)# dpd-interval client 1000`	Configures dead peer detection (DPD).globally for the client or server. • **client**—DPD for the client mode. The default value is 300 (five minutes). • **server**—DPD for the server mode. The default value is 300 (five minutes). • *interval*—Interval, in seconds. The range is from 5 to 3600.
Step 9	**homepage** *homepage-text* **Example:** `Device(config-crypto-ssl-auth-policy)# homepage http://www.abc.com`	Specifies the SSL VPN home page URL.
Step 10	**include-local-lan** **Example:** `Device(config-crypto-ssl-auth-policy)# include-local-lan`	Permits the remote user to access resources on a local LAN, such as a network printer.
Step 11	**ipv6 prefix** *prefix* **Example:** `Device(config-crypto-ssl-auth-policy)# ipv6 prefix 64`	Defines the IPv6 prefix for IPv6 addresses. • *prefix*—Prefix length. The range is from 1 to 128.
Step 12	**keepalive** *seconds* **Example:**	Enables setting the minimum, maximum, and default values, in seconds for keepalive.

	Command or Action	Purpose
	`Device(config-crypto-ssl-auth-policy)# keepalive 500`	
Step 13	**module** *module-name* **Example:** `Device(config-crypto-ssl-auth-policy)# module gina`	Enables the server gateway to download the appropriate module for VPN to connect to a specific group. • **dart**—Downloads the AnyConnect Diagnostic and Reporting Tool (DART) module. • **gina**—Downloads the Start Before Logon (SBL) module.
Step 14	**msie-proxy exception** *exception-name* **Example:** `Device(config-crypto-ssl-auth-policy)# msie-proxy exception 198.51.100.2`	The DNS name or the IP address specified in the *exception-name* argument that must not be sent through the proxy.
Step 15	**msie-proxy option** {**auto** | **bypass** | **none**} **Example:** `Device(config-crypto-ssl-auth-policy)# msie-proxy option bypass`	Specifies the proxy settings for the Microsoft Internet Explorer browser. The proxy settings are required to specify an internal proxy server and to route the browser traffic through the proxy server when connecting to the corporate network. • **auto**—Browser is configured to auto detect proxy server settings. • **bypass**—Local addresses bypass the proxy server. • **none**—Browser is configured to not use the proxy server.
Step 16	**msie-proxy server** {*ip-address* | *dns-name*} **Example:** `Device(config-crypto-ssl-auth-policy)# msie-proxy server 198.51.100.2`	The IP address or the DNS name, optionally followed by the port number of the proxy server. **Note** This command is required if the **msie-proxy option bypass** command is specified.
Step 17	**mtu** *bytes* **Example:** `Device(config-crypto-ssl-auth-policy)# mtu 1000`	(Optional) Enables setting the minimum, maximum, and default MTU value. **Note** The value specified in this command overrides the default MTU specified in the Cisco AnyConnect Secure client configuration. If not specified, the value specified in the Cisco AnyConnect Secure client configuration is the MTU value. If the calculated MTU is less than the MTU specified in this command, this command is ignored.
Step 18	**netmask** *mask* **Example:**	Specifies the netmask of the subnet from which the IP address is assigned to the client. • *mask*—Subnet mask address.

	Command or Action	Purpose
	`Device(config-crypto-ssl-auth-policy)# netmask 255.255.255.0`	

Step 19	Run one of the following commands:
	• **pool** *name*
	• Or
	• **ipv6 pool** *name*
	Example:
	`Device(config-crypto-ssl-auth-policy)# pool abc`
	Example:
	`Device(config-crypto-ssl-auth-policy)# ipv6 pool ipv6pool`
		Defines a local IPv4 or IPv6 address pool for assigning IP addresses to the remote access client.
		• *name*—Name of the local IP address pool.
		Note The local IP address pool must already be defined using the **ip local pool** command.

Step 20	**rekey time** *seconds* **Example:** `Device(config-crypto-ssl-auth-policy)# rekey time 1110`	Specifies the rekey interval, in seconds. The default value is 3600.

Step 21	Run one of the following commands:
	• **route set access-list** *acl-name*
	• Or
	• **ipv6 route set access-list** *access-list-name*
	Example:
	`Device(config-crypto-ssl-auth-policy)# route set access-list acl1`
	Example:
	`Device(config-crypto-ssl-auth-policy)# ipv6 route set access-list acl1`
		Establishes IPv4 or IPv6 routes the access list that must be secured through tunnels.
		• *acl-name*—Access list name.

Step 22	**smartcard-removal-disconnect** **Example:** `Device(config-crypto-ssl-auth-policy)# smartcard-removal-disconnect`	Enables smartcard removal disconnect and specifies that the client should terminate the session when the smart card is removed.
Step 23	**split-dns** *string* **Example:** `Device(config-crypto-ssl-auth-policy)# split-dns example.com example.net`	Allows you to specify up to ten split domain names, which the client should use for private networks.
Step 24	**timeout** {**disconnect** *seconds* | **idle** *seconds* | **session** *seconds*} **Example:** `Device(config-crypto-ssl-auth-policy)# timeout disconnect 10000`	Specifies the timeout, in seconds. • **disconnect** *seconds*—Specifies the retry duration, in seconds, for Cisco AnyConnect client to reconnect to the server gateway. The default value is 0. • **idle** *seconds*—Specifies the idle timeout, in seconds. The default value is 1800 (30 minutes).

	Command or Action	Purpose
		• **session** *seconds*—Specifies the session timeout, in seconds. The default value is 43200 (12 hours).
Step 25	**wins** *primary-server* [*secondary-server*] **Example:** `Device(config-crypto-ssl-auth-policy)# wins 203.0.113.1 203.0.113.115`	Specifies the internal Windows Internet Naming Service (WINS) server addresses. • *primary-server*—IP address of the primary WINS server. • *secondary-server*—(Optional) IP address of the secondary WINS server.
Step 26	**end** **Example:** `Device(config-crypto-ssl-auth-policy)# end`	Exits SSL authorization policy configuration mode and returns to privileged EXEC mode.
Step 27	**show crypto ssl authorization policy** [*policy-name*] **Example:** `Device(config-crypto-ssl-auth-policy)# show crypto ssl authorization policy`	(Optional) Displays the SSL authorization policy.

Verifying SSL VPN Configurations

This section describes how to use **show** commands to verify the SSL VPN configurations:

SUMMARY STEPS

1. **enable**
2. **show crypto ssl proposal** [*name*]
3. **show crypto ssl policy** [*name*]
4. **show crypto ssl profile** [*name*]
5. **show crypto ssl authorization policy** [*name*]
6. **show crypto ssl session** {**user** *user-name* | **profile** *profile-name*}
7. **show crypto ssl stats** [**profile** *profile-name*] [**tunnel**] [**detail**]
8. **clear crypto ssl session** {**profile** *profile-name*| **user** *user-name*}

DETAILED STEPS

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

Enter your password, if prompted.

Step 2 **show crypto ssl proposal** [*name*]

Example:

```
Device# show crypto ssl proposal

SSL Proposal: sslprop
    Protection: 3DES-SHA1
```

Displays the SSL proposal.

Step 3 **show crypto ssl policy** [*name*]

Example:

```
Device# show crypto ssl policy

 SSL Policy: sslpolicy
  Status     : ACTIVE
  Proposal   : sslprop
  IP Address : 10.78.106.23
  Port       : 443
  fvrf       : 0
  Trust Point: TP-self-signed-1183786860
  Redundancy : none
```

Displays the SSL policies.

Step 4 **show crypto ssl profile** [*name*]

Example:

```
Device# show crypto ssl profile

SSL Profile: sslprofile
 Status: ACTIVE
 Match Criteria:
   URL: none
   Policy:
    sslpolicy
 AAA accounting List      : local
 AAA authentication List  :none
 AAA authorization cached  :true
 AAA authorization user List   :default
 AAA authorization user name: sslauth
 AAA authorization group List   :none
 AAA authorization group name: none
 Authentication Mode      : user credentials
 Interface                : SSLVPN-VIF1
   Status: ENABLE
```

Displays the SSL profile.

Step 5 **show crypto ssl authorization policy** [*name*]

Example:

```
Device# show crypto ssl authorization policy

SSL Auth Policy: sslauth
 V4 Parameter:
   Address Pool: SVC_POOL
   Netmask: 255.255.255.0
   Route ACL : split-include
 Banner                  : none
 Home Page               : none
 Idle timeout            : 300
 Disconnect Timeout      : 0
```

```
 Session Timeout         : 43200
 Keepalive Interval      : 0
 DPD Interval            : 300
 Rekey
   Interval: 0
   Method  : none
 Split DNS               : none
 Default domain          : none
 Proxy Settings
     Server: none
     Option: NULL
     Exception(s): none
 Anyconnect Profile Name :
 SBL Enabled             : NO
 MAX MTU                 : 1406
 Smart Card
 Removal Disconnect      : NO
```

Displays the SSL authorization policy.

Step 6 **show crypto ssl session** {**user** *user-name* | **profile** *profile-name*}

Example:

```
Device# show crypto ssl session user LAB

Session Type      : Full Tunnel
Client User-Agent : AnyConnect Windows 3.0.08057

Username          : LAB                 Num Connection : 1
Public IP         : 10.163.209.245
Profile           : sslprofile             Policy Group   : sslauth
Last-Used         : 00:00:02           Created        : *00:58:44.219 PDT Thu Jul 25 2013
Session Timeout   : 43200          Idle Timeout   : 300
DPD GW Timeout    : 300           DPD CL Timeout : 300
Address Pool      : sslvpn-pool     MTU Size       : 1406
Rekey Time        : 0                    Rekey Method   :
Lease Duration    : 43200
Tunnel IP         : 10.1.1.2            Netmask        : 255.255.255.0
Rx IP Packets     : 0                    Tx IP Packets  : 125
CSTP Started      : 00:01:12        Last-Received  : 00:00:02
CSTP DPD-Req sent : 0               Virtual Access : 0
Msie-ProxyServer  : None            Msie-PxyPolicy : Disabled
Msie-Exception    :
Client Ports      : 34552

Device# show crypto ssl session profile sslprofile

SSL profile name: sslprofile
Client_Login_Name  Client_IP_Address  No_of_Connections  Created  Last_Used
LAB                10.163.209.245             1         00:00:33  00:00:00
Error receiving show session info from remote cores
```

Displays SSL VPN session information.

Step 7 **show crypto ssl stats** [**profile** *profile-name*] [**tunnel**] [**detail**]

Example:

```
Device# show crypto ssl stats

SSLVPN Global statistics:
    Active connections       : 0          AAA pending reqs         : 0
    Peak connections         : 1          Peak time                : 1w6d
```

```
    Authentication failures  : 21
    VPN session timeout      : 1          VPN idle timeout         : 0
    User cleared VPN sessions: 0          Login Denined            : 0
    Connect succeed          : 1          Connect failed           : 0
    Reconnect succeed        : 0          Reconnect failed         : 0
    IP Addr Alloc Failed     : 0          VA creation failed       : 0
    Route Insertion Failed   : 0
    IPV6 Addr Alloc Failed   : 0
    IPV6 Route Insert Failed : 0
    IPV6 Hash Insert Failed  : 0
    IPV6 STC Alloc Failed    : 0
    in  CSTP control         : 5          out CSTP control         : 3
    in  CSTP data            : 21         out CSTP data            : 8

Device# show crypto ssl stats tunnel profile prf1
SSLVPN Profile name : prf1
Tunnel Statistics:
    Active connections       : 0
    Peak connections         : 0          Peak time                : never
    Connect succeed          : 0          Connect failed           : 0
    Reconnect succeed        : 0          Reconnect failed         : 0
    DPD timeout              : 0
  Client
    in  CSTP frames          : 0          in  CSTP control         : 0
    in  CSTP data            : 0          in  CSTP bytes           : 0
    out CSTP frames          : 0          out CSTP control         : 0
    out CSTP data            : 0          out CSTP bytes           : 0
    cef in  CSTP data frames : 0          cef in  CSTP data bytes  : 0
    cef out CSTP data frames : 0          cef out CSTP data bytes  : 0
  Server
    In  IP pkts              : 0          In  IP bytes             : 0
    Out IP pkts              : 0          Out IP bytes             : 0
```

Displays SSL VPN statistics.

Step 8 **clear crypto ssl session** {**profile** *profile-name*| **user** *user-name*}

Example:

```
Device# clear crypto ssl session sslprofile
```

Clears SSL VPN session.

Configuration Examples for SSL VPN

Example: Creating a Virtual Template for SSL VPN

The following example shows how to create a template for SSL VPN:

```
Device> enable
Device# configure terminal
Device(config)# interface virtual-template 1 type vpn
Device(config-if)# ip unnumbered Te0/0/4
Device(config-if)# ip tcp adjust-mss 1300
Device(config-if)# end
```

Example: Specifying the AnyConnect Image and Profile

The following example shows how to specify the Cisco AnyConnect image and profile:

```
Device> enable
Device# configure terminal
Device(config)# crypto vpn anyconnect bootflash:/webvpn/anyconnect-win-3.1.04072-k9.pkg
sequence 1
Device(config)# crypto vpn anyconnect profile Employee bootflash:/Employee.xml
Device(config)# end
```

Example: Configuring an SSL Proposal

The following example shows how to configure an SSL proposal:

```
Device> enable
Device# configure terminal
Device(config)# crypto ssl proposal proposal1
Device(config-crypto-ssl-proposal)# protection rsa-3des-ede-sha1 rsa-aes128-sha1
Device(config-crypto-ssl-proposal)# end
```

Example: Configuring an SSL Policy

The following example shows how to configure an SSL policy:

```
Device> enable
Device# configure terminal
Device(config)# crypto ssl policy policy1
Device(config-crypto-ssl-policy)# ip address local 10.0.0.1 port 443
Device(config-crypto-ssl-policy)# pki trustpoint tp1 sign
Device(config-crypto-ssl-policy)# ssl proposal proposal1
Device(config-crypto-ssl-policy)# no shut
Device(config-crypto-ssl-policy)# end
```

Example: Configuring an SSL Profile

The following example shows how to configure an SSL profile:

```
Device> enable
Device# configure terminal
Device(config)# crypto ssl profile profile1
Device(config-crypto-ssl-profile)# aaa accounting user-pass list list1
Device(config-crypto-ssl-profile)# aaa authentication user-pass list list2
Device(config-crypto-ssl-profile)# aaa authorization group override user-pass list list1
user1
Device(config-crypto-ssl-profile)# aaa authorization user user-pass list list1 user1
Device(config-crypto-ssl-profile)# match policy policy1
Device(config-crypto-ssl-profile)# match url www.abc.com
Device(config-crypto-ssl-profile)# virtual-template 1
Device(config-crypto-ssl-profile)# no shut
Device(config-crypto-ssl-profile)# end
```

Example: Configuring an SSL Authorization Policy

The following example shows how to configure an SSL authorization policy:

```
Device> enable
Device# configure terminal
Device(config)# crypto ssl authorization policy policy1
Device(config-crypto-ssl-auth-policy)# banner This is SSL VPN tunnel.
Device(config-crypto-ssl-auth-policy)# client profile Employee
Device(config-crypto-ssl-auth-policy)# def-domain cisco
Device(config-crypto-ssl-auth-policy)# dns 198.51.100.1 198.51.100.100
Device(config-crypto-ssl-auth-policy)# dpd client 1000
Device(config-crypto-ssl-auth-policy)# homepage http://www.abc.com
Device(config-crypto-ssl-auth-policy)# include-local-lan
Device(config-crypto-ssl-auth-policy)# keepalive 500
Device(config-crypto-ssl-auth-policy)# module gina
Device(config-crypto-ssl-auth-policy)# msie-proxy exception 198.51.100.2
Device(config-crypto-ssl-auth-policy)# msie-proxy option bypass
Device(config-crypto-ssl-auth-policy)# msie-proxy server 198.51.100.2
Device(config-crypto-ssl-auth-policy)# mtu 1000
Device(config-crypto-ssl-auth-policy)# netmask 255.255.255.0
Device(config-crypto-ssl-auth-policy)# pool abc
Device(config-crypto-ssl-auth-policy)# rekey interval 1110
Device(config-crypto-ssl-auth-policy)# route set access-list acl1
Device(config-crypto-ssl-auth-policy)# smartcard-removal-disconnect
Device(config-crypto-ssl-auth-policy)# split-dns abc1
Device(config-crypto-ssl-auth-policy)# timeout disconnect 10000
Device(config-crypto-ssl-auth-policy)# wins 203.0.113.1 203.0.113.115
Device(config-crypto-ssl-auth-policy)# end
```

The following example shows how to enable IPv6 support for SSL VPN:

```
Device> enable
Device# configure terminal
Device(config)# crypto ssl authorization policy policy1
Device(config-crypto-ssl-auth-policy)# banner This is SSL VPN tunnel.
Device(config-crypto-ssl-auth-policy)# client profile profile1
Device(config-crypto-ssl-auth-policy)# def-domain cisco
Device(config-crypto-ssl-auth-policy)# ipv6 dns 2001:DB8:1::1 2001:DB8:2::2
Device(config-crypto-ssl-auth-policy)# dpd client 1000
Device(config-crypto-ssl-auth-policy)# homepage http://www.abc.com
Device(config-crypto-ssl-auth-policy)# include-local-lan
Device(config-crypto-ssl-auth-policy)# ipv6 prefix 64
Device(config-crypto-ssl-auth-policy)# ipv6 route set access-list acl1
Device(config-crypto-ssl-auth-policy)# keepalive 500
Device(config-crypto-ssl-auth-policy)# module gina
Device(config-crypto-ssl-auth-policy)# msie-proxy exception 198.51.100.2
Device(config-crypto-ssl-auth-policy)# msie-proxy option bypass
Device(config-crypto-ssl-auth-policy)# msie-proxy server 198.51.100.2
Device(config-crypto-ssl-auth-policy)# mtu 1000
Device(config-crypto-ssl-auth-policy)# ipv6 pool ipv6pool
Device(config-crypto-ssl-auth-policy)# rekey interval 1110
Device(config-crypto-ssl-auth-policy)# route set access-list acl1
Device(config-crypto-ssl-auth-policy)# smartcard-removal-disconnect
Device(config-crypto-ssl-auth-policy)# split-dns abc1
Device(config-crypto-ssl-auth-policy)# timeout disconnect 10000
Device(config-crypto-ssl-auth-policy)# wins 203.0.113.1 203.0.113.115
Device(config-crypto-ssl-auth-policy)# end
```

Additional References for SSL VPN

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
Recommended cryptographic algorithms	Next Generation Encryption

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 SSL VPN

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 31: Feature Information for SSL VPN

Feature Name	Release	Feature Information
SSL VPN	Cisco IOS XE Release 17.7.1a	The SSL VPN feature is introduced. This feature provides support in the Cisco IOS XE software for remote user access to enterprise networks from anywhere on the internet.

CHAPTER 21

Configuring Call Home

The Call Home feature provides e-mail-based and web-based notification of critical system events. A versatile range of message formats are available for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications. Common uses of this feature may include direct paging of a network support engineer, e-mail notification to a Network Operations Center, XML delivery to a support website, and use of Cisco Smart Call Home services for direct case generation with the Cisco Systems Technical Assistance Center (TAC).

This chapter describes how to configure the Call Home feature in Cisco IOS Release 15.4(3)S and later releases for the Cisco ISR 4400 Series and Cisco ISR 4300 Series Routers.

This chapter includes the following sections:

Finding Feature Information

Your software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

Use the Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, see http://tools.cisco.com/ITDIT/CFN/. A Cisco account is not required to access the Cisco Feature Navigator.

Prerequisites for Call Home

The following are the prerequisites before you configure Call Home:

- Contact e-mail address (required for full registration with Smart Call Home, optional if Call Home is enabled in anonymous mode), phone number (optional), and street address information (optional) should be configured so that the receiver can determine the origin of messages received.
- At least one destination profile (predefined or user-defined) must be configured. The destination profile you use depends on whether the receiving entity is a pager, an e-mail address, or an automated service such as Cisco Smart Call Home.

  If the destination profile uses e-mail message delivery, you must specify a Simple Mail Transfer Protocol (SMTP) server.
- The router must have IP connectivity to an e-mail server or the destination HTTP server.
- If Cisco Smart Call Home is used, an active service contract covering the device is required to provide full Cisco Smart Call Home service.

Information About Call Home

The Call Home feature can deliver alert messages containing information on configuration, environmental conditions, inventory, syslog, snapshot, and crash events. It provides these alert messages as either e-mail-based or web-based messages. Multiple message formats are available, allowing for compatibility with pager services, standard e-mail, or XML-based automated parsing applications. This feature can deliver alerts to multiple recipients, referred to as Call Home destination profiles, each with configurable message formats and content categories. A predefined destination profile is provided for sending alerts to the Cisco TAC (callhome@cisco.com). You can also define your own destination profiles.

Flexible message delivery and format options make it easy to integrate specific support requirements.

This section contains the following subsections:

- Benefits of Using Call Home
- Obtaining Smart Call Home Services

Benefits of Using Call Home

The Call Home feature offers the following benefits:

- Multiple message-format options, which include:
  - Short Text—Suitable for pagers or printed reports.
  - Plain Text—Full formatted message information suitable for human reading.
  - XML—Machine-readable format using XML and Adaptive Markup Language (AML) document type definitions (DTDs). The XML format enables communication with the Cisco TAC.
- Multiple concurrent message destinations.
- Multiple message categories including configuration, environmental conditions, inventory, syslog, snapshot, and crash events.
- Filtering of messages by severity and pattern matching.

- Scheduling of periodic message sending.

Obtaining Smart Call Home Services

If you have a service contract directly with Cisco, you can register for the Smart Call Home service. Smart Call Home analyzes Smart Call Home messages and provides background information and recommendations. For known issues, particularly online diagnostics failures, Automatic Service Requests are generated with the Cisco TAC.

Smart Call Home offers the following features:

- Continuous device health monitoring and real-time diagnostic alerts.
- Analysis of Smart Call Home messages and, if needed, Automatic Service Request generation routed to the correct TAC team, including detailed diagnostic information to speed problem resolution.
- Secure message transport directly from your device or through an HTTP proxy server or a downloadable Transport Gateway (TG). You can use a TG aggregation point to support multiple devices or in cases where security dictates that your devices may not be connected directly to the Internet.
- Web-based access to Smart Call Home messages and recommendations, inventory, and configuration information for all Smart Call Home devices provides access to associated field notices, security advisories, and end-of-life information.

You need the following items to register for Smart Call Home:

- SMARTnet contract number for your router
- Your e-mail address
- Your Cisco.com username

For more information about Smart Call Home, see https://supportforums.cisco.com/community/4816/smart-call-home.

Anonymous Reporting

Smart Call Home is a service capability included with many Cisco service contracts and is designed to assist customers resolve problems more quickly. In addition, the information gained from crash messages helps Cisco understand equipment and issues occurring in the field. If you decide not to use Smart Call Home, you can still enable Anonymous Reporting to allow Cisco to securely receive minimal error and health information from the device. If you enable Anonymous Reporting, your customer identity will remain anonymous, and no identifying information will be sent.

Note When you enable Anonymous Reporting, you acknowledge your consent to transfer the specified data to Cisco or to vendors operating on behalf of Cisco (including countries outside the United States). Cisco maintains the privacy of all customers. For information about how Cisco treats personal information, see the Cisco Privacy Statement at http://www.cisco.com/web/siteassets/legal/privacy.html.

When Call Home is configured in an anonymous way, only crash, inventory, and test messages are sent to Cisco. No customer identifying information is sent.

For more information about what is sent in these messages, see Alert Group Trigger Events and Commands, on page 340.

How to Configure Call Home

The following sections show how to configure Call Home using a single command:

The following sections show detailed or optional configurations:

Configuring Smart Call Home (Single Command)

To enable all Call Home basic configurations using a single command, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home reporting** {**anonymous** | **contact-email-addr** *email-address*} [**http-proxy** {*ipv4-address* | *ipv6-address* | *name*} **port** *port-number*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.

	Command or Action	Purpose
Step 2	**call-home reporting** {**anonymous** | **contact-email-addr** *email-address*} [**http-proxy** {*ipv4-address* | *ipv6-address* | *name*} **port** *port-number*] **Example:** `Router(config)# call-home reporting contact-email-addr email@company.com`	Enables the basic configurations for Call Home using a single command. • **anonymous**—Enables Call-Home TAC profile to send only crash, inventory, and test messages and send the messages anonymously. • **contact-email-addr**—Enables Smart Call Home service full reporting capability and sends a full inventory message from Call-Home TAC profile to Smart Call Home server to start full registration process. • **http-proxy** {*ipv4-address* | *ipv6-address* | *name*}—Configures an ipv4 or ipv6 address or server name. Maximum length is 64 characters. • **port** *port-number*—Port number. Range is 1 to 65535. **Note** The HTTP proxy option allows you to make use of your own proxy server to buffer and secure Internet connections from your devices. **Note** After successfully enabling Call Home either in anonymous or full registration mode using the **call-home reporting** command, an inventory message is sent out. If Call Home is enabled in full registration mode, a Full Inventory message for full registration mode is sent out. If Call Home is enabled in anonymous mode, an anonymous inventory message is sent out. For more information about what is sent in these messages, see Alert Group Trigger Events and Commands, on page 340.

Configuring and Enabling Smart Call Home

For application and configuration information about the Cisco Smart Call Home service, see the "Getting Started" section of the Smart Call Home User Guide at https://supportforums.cisco.com/community/4816/smart-call-home. This document includes configuration examples for sending Smart Call Home messages directly from your device or through a transport gateway (TG) aggregation point.

Note For security reasons, we recommend that you use the HTTPS transport options, due to the additional payload encryption that HTTPS offers. The Transport Gateway software is downloadable from Cisco.com and is available if you require an aggregation point or a proxy for connection to the Internet.

Enabling and Disabling Call Home

To enable or disable the Call Home feature, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **service call-home**
3. **no service call-home**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**service call-home** **Example:** `Router(config)# service call-home`	Enables the Call Home feature.
Step 3	**no service call-home** **Example:** `Router(config)# no service call-home`	Disables the Call Home feature.

Configuring Contact Information

Each router must include a contact e-mail address (except if Call Home is enabled in anonymous mode). You can optionally include a phone number, street address, contract ID, customer ID, and site ID.

To assign the contact information, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **contact-email-addr** *email-address*
4. **phone-number** +*phone-number*
5. **street-address** *street-address*
6. **customer-id** *text*
7. **site-id** *text*
8. **contract-id** *text*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters the Call Home configuration submode.
Step 3	**contact-email-addr** *email-address* **Example:** `Router(cfg-call-home)# contact-email-addr username@example.com`	Designates your e-mail address. Enter up to 200 characters in e-mail address format with no spaces.
Step 4	**phone-number** +*phone-number* **Example:** `Router(cfg-call-home)# phone-number +1-800-555-4567`	(Optional) Assigns your phone number. **Note** The number must begin with a plus (+) prefix and may contain only dashes (-) and numbers. Enter up to 17 characters. If you include spaces, you must enclose your entry in quotes (“”).
Step 5	**street-address** *street-address* **Example:** `Router(cfg-call-home)# street-address “1234 Picaboo Street, Any city, Any state, 12345‟`	(Optional) Assigns your street address where RMA equipment can be shipped. Enter up to 200 characters. If you include spaces, you must enclose your entry in quotes (“”).
Step 6	**customer-id** *text* **Example:** `Router(cfg-call-home)# customer-id Customer1234`	(Optional) Identifies customer ID. Enter up to 64 characters. If you include spaces, you must enclose your entry in quotes (“”).
Step 7	**site-id** *text* **Example:** `Router(cfg-call-home)# site-id Site1ManhattanNY`	(Optional) Identifies customer site ID. Enter up to 200 characters. If you include spaces, you must enclose your entry in quotes (“”).
Step 8	**contract-id** *text* **Example:** `Router(cfg-call-home)# contract-id Company1234`	(Optional) Identifies your contract ID for the router. Enter up to 64 characters. If you include spaces, you must enclose your entry in quotes (“”).

Example

The following example shows how to configure contact information:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# call-home
Router(cfg-call-home)# contact-email-addr username@example.com
```

```
Router(cfg-call-home)# phone-number +1-800-555-4567
Router(cfg-call-home)# street-address "1234 Picaboo Street, Any city, Any state, 12345"
Router(cfg-call-home)# customer-id Customer1234
Router(cfg-call-home)# site-id Site1ManhattanNY
Router(cfg-call-home)# contract-id Company1234
Router(cfg-call-home)# exit
```

Configuring Destination Profiles

A destination profile contains the required delivery information for an alert notification. At least one destination profile is required. You can configure multiple destination profiles of one or more types.

You can create and define a new destination profile or copy and use the predefined destination profile. If you define a new destination profile, you must assign a profile name.

> **Note** If you use the Cisco Smart Call Home service, the destination profile must use the XML message format.

You can configure the following attributes for a destination profile:

- Profile name—String that uniquely identifies each user-defined destination profile. The profile name is limited to 31 characters and is not case-sensitive.

  > **Note** You cannot use **all** as a profile name.

- Transport method—Transport mechanism, either e-mail or HTTP (including HTTPS), for delivery of alerts.
  - For user-defined destination profiles, e-mail is the default, and you can enable either or both transport mechanisms. If you disable both methods, e-mail is enabled.
  - For the predefined Cisco TAC profile, you can enable either transport mechanism, but not both.
- Destination address—The actual address related to the transport method to which the alert should be sent.
- Message formatting—The message format used for sending the alert. The format options for a user-defined destination profile are long-text, short-text, or XML. The default is XML. For the predefined Cisco TAC profile, only XML is allowed.
- Message size—The maximum destination message size. The valid range is 50 to 3,145,728 Bytes. The default is 3,145,728 Bytes.

  Anonymous reporting—You can choose for your customer identity to remain anonymous, and no identifying information is sent.
- Subscribing to interesting alert-groups—You can choose to subscribe to alert-groups highlighting your interests.

This section contains the following subsections:

- Creating a New Destination Profile, on page 309

- Copying a Destination Profile, on page 310
- Setting Profiles to Anonymous Mode, on page 311

Creating a New Destination Profile

To create and configure a new destination profile, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **profile** *name*
4. [**no**] **destination transport-method** {**email** | **http**}
5. **destination address** {**email** *email-address* | **http** *url*}
6. **destination preferred-msg-format** {**long-text** | **short-text** | **xml**}
7. **destination message-size-limit** *bytes*
8. **active**
9. **end**
10. **show call-home profile** {*name* | **all**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters the Call Home configuration submode.
Step 3	**profile** *name* **Example:** `Router(config-call-home)# profile profile1`	Enters the Call Home destination profile configuration submode for the specified destination profile. If the specified destination profile does not exist, it is created.
Step 4	[**no**] **destination transport-method** {**email** | **http**} **Example:** `Router(cfg-call-home-profile)# destination transport-method email`	(Optional) Enables the message transport method. The **no** option disables the method.
Step 5	**destination address** {**email** *email-address* | **http** *url*} **Example:** `Router(cfg-call-home-profile)# destination address email myaddress@example.com`	Configures the destination e-mail address or URL to which Call Home messages are sent. **Note** When entering a destination URL, include either **http://** or **https://**, depending on whether the server is a secure server.

	Command or Action	Purpose
Step 6	**destination preferred-msg-format** {**long-text** | **short-text** | **xml**} **Example:** `Router(cfg-call-home-profile)# destination preferred-msg-format xml`	(Optional) Configures a preferred message format. The default is XML.
Step 7	**destination message-size-limit** *bytes* **Example:** `Router(cfg-call-home-profile)# destination message-size-limit 3145728`	(Optional) Configures a maximum destination message size for the destination profile.
Step 8	**active** **Example:** `Router(cfg-call-home-profile)# active`	Enables the destination profile. By default, the profile is enabled when it is created.
Step 9	**end** **Example:** `Router(cfg-call-home-profile)# end`	Returns to privileged EXEC mode.
Step 10	**show call-home profile** {*name* | **all**} **Example:** `Router# show call-home profile profile1`	Displays the destination profile configuration for the specified profile or all configured profiles.

Copying a Destination Profile

To create a new destination profile by copying an existing profile, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **copy profile** *source-profile target-profile*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters the Call Home configuration submode.

	Command or Action	Purpose
Step 3	**copy profile** *source-profile target-profile* **Example:** `Router(cfg-call-home)# copy profile profile1 profile2`	Creates a new destination profile with the same configuration settings as the existing destination profile.

Setting Profiles to Anonymous Mode

To set an anonymous profile, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **profile** *name*
4. **anonymous-reporting-only**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters the Call Home configuration submode.
Step 3	**profile** *name* **Example:** `Router(cfg-call-home) profile Profile-1`	Enables the profile configuration mode.
Step 4	**anonymous-reporting-only** **Example:** `Router(cfg-call-home-profile)# anonymous-reporting-only`	Sets the profile to anonymous mode. **Note** By default, Call Home sends a full report of all types of events subscribed in the profile. When **anonymous-reporting-only** is set, only crash, inventory, and test messages will be sent.

Subscribing to Alert Groups

An alert group is a predefined subset of Call Home alerts supported in all routers. Different types of Call Home alerts are grouped into different alert groups depending on their type. The following alert groups are available:

- Crash

- Configuration
- Environment
- Inventory
- Snapshot
- Syslog

This section contains the following subsections:

- Periodic Notification, on page 314
- Message Severity Threshold, on page 315
- Configuring a Snapshot Command List, on page 315

The triggering events for each alert group are listed in Alert Group Trigger Events and Commands, on page 340, and the contents of the alert group messages are listed in Message Contents, on page 347.

You can select one or more alert groups to be received by a destination profile.

Note A Call Home alert is only sent to destination profiles that have subscribed to the alert group containing that Call Home alert. In addition, the alert group must be enabled.

To subscribe a destination profile to one or more alert groups, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **alert-group** {**all** | **configuration** | **environment** | **inventory** | **syslog** | **crash** | **snapshot**}
4. **profile** *name*
5. **subscribe-to-alert-group all**
6. **subscribe-to-alert-group configuration** [**periodic** {**daily** *hh:mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}]
7. **subscribe-to-alert-group environment** [**severity** {**catastrophic** | **disaster** | **fatal** | **critical** | **major** | **minor** | **warning** | **notification** | **normal** | **debugging**}]
8. **subscribe-to-alert-group inventory** [**periodic** {**daily** *hh:mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}]
9. **subscribe-to-alert-group syslog** [**severity** {**catastrophic** | **disaster** | **fatal** | **critical** | **major** | **minor** | **warning** | **notification** | **normal** | **debugging**}]
10. **subscribe-to-alert-group crash**
11. **subscribe-to-alert-group snapshot periodic** {**daily** *hh:mm* | **hourly** *mm* | **interval** *mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}
12. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**alert-group** {**all** | **configuration** | **environment** | **inventory** | **syslog** | **crash** | **snapshot**} **Example:** `Router(cfg-call-home)# alert-group all`	Enables the specified alert group. Use the keyword **all** to enable all alert groups. By default, all alert groups are enabled.
Step 4	**profile** *name* **Example:** `Router(cfg-call-home)# profile profile1`	Enters the Call Home destination profile configuration submode for the specified destination profile.
Step 5	**subscribe-to-alert-group all** **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group all`	Subscribes to all available alert groups using the lowest severity. You can subscribe to alert groups individually by specific type, as described in Step 6 through Step 11. **Note** This command subscribes to the syslog debug default severity. This causes a large number of syslog messages to generate. You should subscribe to alert groups individually, using appropriate severity levels and patterns when possible.
Step 6	**subscribe-to-alert-group configuration** [**periodic** {**daily** *hh:mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group configuration periodic daily 12:00`	Subscribes this destination profile to the Configuration alert group. The Configuration alert group can be configured for periodic notification, as described in Periodic Notification, on page 314.
Step 7	**subscribe-to-alert-group environment** [**severity** {**catastrophic** | **disaster** | **fatal** | **critical** | **major** | **minor** | **warning** | **notification** | **normal** | **debugging**}] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group environment severity major`	Subscribes this destination profile to the Environment alert group. The Environment alert group can be configured to filter messages based on severity, as described in Message Severity Threshold, on page 315.

	Command or Action	Purpose
Step 8	**subscribe-to-alert-group inventory** [**periodic** {**daily** *hh:mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group inventory periodic monthly 1 12:00`	Subscribes this destination profile to the Inventory alert group. The Inventory alert group can be configured for periodic notification, as described in Periodic Notification, on page 314.
Step 9	**subscribe-to-alert-group syslog** [**severity** {**catastrophic** | **disaster** | **fatal** | **critical** | **major** | **minor** | **warning** | **notification** | **normal** | **debugging**}] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group environment severity major`	Subscribes this destination profile to the Syslog alert group. The Syslog alert group can be configured to filter messages based on severity, as described in Message Severity Threshold, on page 315. You can specify a text pattern to be matched within each syslog message. If you configure a pattern, a Syslog alert group message is sent only if it contains the specified pattern and meets the severity threshold. If the pattern contains spaces, you must enclose it in quotes (“”). You can specify up to five patterns for each destination profile.
Step 10	**subscribe-to-alert-group crash** **Example:** `Router(cfg-call-home-profile)# [no | default] subscribe-to-alert-group crash`	Subscribes to the Crash alert group in user profile. By default, TAC profile subscribes to the Crash alert group and cannot be unsubscribed.
Step 11	**subscribe-to-alert-group snapshot periodic** {**daily** *hh:mm* | **hourly** *mm* | **interval** *mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*} **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group snapshot periodic daily 12:00`	Subscribes this destination profile to the Snapshot alert group. The Snapshot alert group can be configured for periodic notification, as described in Periodic Notification, on page 314. By default, the Snapshot alert group has no command to run. You can add commands into the alert group, as described in Configuring a Snapshot Command List, on page 315. In doing so, the output of the commands added in the Snapshot alert group will be included in the snapshot message.
Step 12	**exit** **Example:** `Router(cfg-call-home-profile)# exit`	Exits the Call Home destination profile configuration submode.

Periodic Notification

When you subscribe a destination profile to the Configuration, Inventory, or Snapshot alert group, you can choose to receive the alert group messages asynchronously or periodically at a specified time. The sending period can be one of the following:

- Daily—Specifies the time of day to send, using an hour:minute format *hh:mm*, with a 24-hour clock (for example, 14:30).

- Weekly—Specifies the day of the week and time of day in the format *day hh:mm*, where the day of the week is spelled out (for example, Monday).
- Monthly—Specifies the numeric date, from 1 to 31, and the time of day, in the format *date hh:mm*.
- Interval—Specifies the interval at which the periodic message is sent, from 1 to 60 minutes.
- Hourly—Specifies the minute of the hour at which the periodic message is sent, from 0 to 59 minutes.

Note Hourly and by interval periodic notifications are available for the Snapshot alert group only.

Message Severity Threshold

When you subscribe a destination profile to the Environment or Syslog alert group, you can set a threshold for the sending of alert group messages based on the level of severity of the message. Any message with a value lower than the destination profile specified threshold is not sent to the destination.

The severity threshold is configured using the keywords listed in the following table. The severity threshold ranges from catastrophic (level 9, highest level of urgency) to debugging (level 0, lowest level of urgency). If no severity threshold is configured for the Syslog or Environment alert groups, the default is debugging (level 0). The Configuration and Inventory alert groups do not allow severity configuration; severity is always set as normal.

Note Call Home severity levels are not the same as system message logging severity levels.

Table 32: Severity and Syslog Level Mapping

Level	Keyword	Syslog Level	Description
9	catastrophic	—	Network-wide catastrophic failure.
8	disaster	—	Significant network impact.
7	fatal	Emergency (0)	System is unusable.
6	critical	Alert (1)	Critical conditions, immediate attention needed.
5	major	Critical (2)	Major conditions.
4	minor	Error (3)	Minor conditions.
3	warning	Warning (4)	Warning conditions.
2	notification	Notice (5)	Basic notification and informational messages. Possibly independently insignificant.
1	normal	Information (6)	Normal event signifying return to normal state.
0	debugging	Debug (7)	Debugging messages.

Configuring a Snapshot Command List

To configure a snapshot command list, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. [**no** | **default**] **alert-group-config snapshot**
4. [**no** | **default**] **add-command** *command string*
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	[**no** | **default**] **alert-group-config snapshot** **Example:** `Router(cfg-call-home)# alert-group-config snapshot`	Enters snapshot configuration mode. The **no** or **default** command will remove all snapshot command.
Step 4	[**no** | **default**] **add-command** *command string* **Example:** `Router(cfg-call-home-snapshot)# add-command "show version"`	Adds the command to the Snapshot alert group. The **no** or **default** command removes the corresponding command. • *command string*—IOS command. Maximum length is 128.
Step 5	**exit** **Example:** `Router(cfg-call-home-snapshot)# exit`	Exits and saves the configuration.

Configuring General E-Mail Options

To use the e-mail message transport, you must configure at least one Simple Mail Transfer Protocol (SMTP) e-mail server address. You can configure the from and reply-to e-mail addresses, and you can specify up to four backup e-mail servers.

Note the following guidelines when configuring general e-mail options:

- Backup e-mail servers can be defined by repeating the **mail-server** command using different priority numbers.
- The **mail-server priority** number parameter can be configured from 1 to 100. The server with the highest priority (lowest priority number) is tried first.

To configure general e-mail options, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **mail-server** [{*ipv4-address* | *ipv6-address*} | *name*] **priority** *number*
4. **sender from** *email-address*
5. **sender reply-to** *email-address*
6. **source-interface** *interface-name*
7. **vrf** *vrf-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**mail-server** [{*ipv4-address* | *ipv6-address*} | *name*] **priority** *number* **Example:** `Router(cfg-call-home)# mail-server stmp.example.com priority 1`	Assigns an e-mail server address and its relative priority among configured e-mail servers. Provide either of these: • The e-mail server's IP address. • The e-mail server's fully qualified domain name (FQDN) of 64 characters or less. Assign a priority number between 1 (highest priority) and 100 (lowest priority).
Step 4	**sender from** *email-address* **Example:** `Router(cfg-call-home)# sender from username@example.com`	(Optional) Assigns the e-mail address that appears in the from field in Call Home e-mail messages. If no address is specified, the contact e-mail address is used.
Step 5	**sender reply-to** *email-address* **Example:** `Router(cfg-call-home)# sender reply-to username@example.com`	(Optional) Assigns the e-mail address that appears in the reply-to field in Call Home e-mail messages.
Step 6	**source-interface** *interface-name* **Example:** `Router(cfg-call-home)# source-interface loopback1`	Assigns the source interface name to send call-home messages. • *interface-name*—Source interface name. Maximum length is 64.

	Command or Action	Purpose
		Note For HTTP messages, use the **ip http client source-interface** *interface-name* command in global configuration mode to configure the source interface name. This allows all HTTP clients on the device to use the same source interface.
Step 7	**vrf** *vrf-name* **Example:** `Router(cfg-call-home)# vrf vpn1`	(Optional) Specifies the VRF instance to send call-home e-mail messages. If no vrf is specified, the global routing table is used. **Note** For HTTP messages, if the source interface is associated with a VRF, use the **ip http client source-interface** *interface-name* command in global configuration mode to specify the VRF instance that will be used for all HTTP clients on the device.

Example

The following example shows the configuration of general e-mail parameters, including a primary and secondary e-mail server:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# call-home
Router(cfg-call-home)# mail-server smtp.example.com priority 1
Router(cfg-call-home)# mail-server 192.168.0.1 priority 2
Router(cfg-call-home)# sender from username@example.com
Router(cfg-call-home)# sender reply-to username@example.com
Router(cfg-call-home)# source-interface loopback1
Router(cfg-call-home)# vrf vpn1
Router(cfg-call-home)# exit
Router(config)#
```

Specifying Rate Limit for Sending Call Home Messages

To specify the rate limit for sending Call Home messages, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **rate-limit** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**rate-limit** *number* **Example:** `Router(cfg-call-home)# rate-limit 40`	Specifies a limit on the number of messages sent per minute. • *number*—Range is 1 to 60. The default is 20.

Specifying HTTP Proxy Server

To specify an HTTP proxy server for sending Call Home HTTP(S) messages to a destination, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **http-proxy** {*ipv4-address* | *ipv6-address* | *name*} **port** *port-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**http-proxy** {*ipv4-address* | *ipv6-address* | *name*} **port** *port-number* **Example:** `Router(cfg-call-home)# http-proxy 192.0.2.1 port 1`	Specifies the proxy server for the HTTP request.

Enabling AAA Authorization to Run IOS Commands for Call Home Messages

To specify an HTTP proxy server for sending Call Home HTTP(S) messages to a destination, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **aaa-authorization**
4. **aaa-authorization** [**username** *username*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**aaa-authorization** **Example:** `Router(cfg-call-home)# aaa-authorization`	Enables AAA authorization. **Note** By default, AAA authorization is disabled for Call Home.
Step 4	**aaa-authorization** [**username** *username*] **Example:** `Router(cfg-call-home)# aaa-authorization username user`	Specifies the username for authorization. • **username** *username*—Default username is callhome. Maximum length is 64.

Configuring Syslog Throttling

To specify an HTTP proxy server for sending Call Home HTTP(S) messages to a destination, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. [**no**] **syslog-throttling**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	[**no**] **syslog-throttling** **Example:** `Router(cfg-call-home)# syslog-throttling`	Enables or disables call-home syslog message throttling and avoids sending repetitive call-home syslog messages. **Note** By default, syslog message throttling is enabled.

Configuring Call Home Data Privacy

The data-privacy command scrubs data, such as IP addresses, from running configuration files to protect the privacy of customers. Enabling the data-privacy command can affect CPU utilization when scrubbing a large amount of data. Currently, the **show** command output is not being scrubbed except for configuration messages in the outputs for the **show running-config all** and the**show startup-config data** commands.

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **data-privacy** {**level** {**normal** | **high**} | **hostname**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**data-privacy** {**level** {**normal** | **high**} | **hostname**} **Example:** `Router(cfg-call-home)# data-privacy level high`	Scrubs data from running configuration file to protect the privacy of the user. The default data-privacy level is normal. **Note** Enabling the data-privacy command can affect CPU utilization when scrubbing a large amount of data. • **normal**—Scrubs all normal-level commands.

	Command or Action	Purpose
		• **high**—Scrubs all normal-level commands plus the IP domain name and IP address commands. • **hostname**—Scrubs all high-level commands plus the hostname command. **Note** Scrubbing the hostname from configuration messages can cause Smart Call Home processing failure on some platforms.

Sending Call Home Communications Manually

You can manually send several types of Call Home communications. To send Call Home communications, perform the tasks in this section. This section contains the following subsections:

- Sending a Call Home Test Message Manually, on page 322
- Sending Call Home Alert Group Messages Manually, on page 322
- Submitting Call Home Analysis and Report Requests, on page 323
- Manually Sending Command Output Message for One Command or a Command List, on page 325

Sending a Call Home Test Message Manually

You can use the **call-home test** command to send a user-defined Call Home test message.

To manually send a Call Home test message, perform the following step:

SUMMARY STEPS

1. **call-home test** [*"test-message"*] **profile** *name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**call-home test** [*"test-message"*] **profile** *name* **Example:** `Router# call-home test profile profile1`	Sends a test message to the specified destination profile. The user-defined test message text is optional but must be enclosed in quotes ("") if it contains spaces. If no user-defined message is configured, a default message is sent.

Sending Call Home Alert Group Messages Manually

You can use the **call-home send** command to manually send a specific alert group message.

Note the following guidelines when manually sending a Call Home alert group message:

- Only the crash, snapshot, configuration, and inventory alert groups can be sent manually.

- When you manually trigger a crash, snapshot, configuration, or inventory alert group message and you specify a destination profile name, a message is sent to the destination profile regardless of the profile's active status, subscription status, or severity setting.
- When you manually trigger a crash, snapshot, configuration, or inventory alert group message and do not specify a destination profile name, a message is sent to all active profiles that have either a normal or periodic subscription to the specified alert group.

To manually trigger Call Home alert group messages, perform the following steps:

SUMMARY STEPS

1. **call-home send alert-group snapshot** [**profile** *name*]
2. **call-home send alert-group crash** [**profile** *name*]
3. **call-home send alert-group configuration** [**profile** *name*]
4. **call-home send alert-group inventory** [**profile** *name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**call-home send alert-group snapshot** [**profile** *name*] **Example:** `Router# call-home send alert-group snapshot profile profile1`	Sends a snapshot alert group message to one destination profile if specified, or to all subscribed destination profiles.
Step 2	**call-home send alert-group crash** [**profile** *name*] **Example:** `Router# call-home send alert-group crash profile profile1`	Sends a crash alert group message to one destination profile if specified, or to all subscribed destination profiles.
Step 3	**call-home send alert-group configuration** [**profile** *name*] **Example:** `Router# call-home send alert-group configuration profile profile1`	Sends a configuration alert group message to one destination profile if specified, or to all subscribed destination profiles.
Step 4	**call-home send alert-group inventory** [**profile** *name*] **Example:** `Router# call-home send alert-group inventory profile profile1`	Sends an inventory alert group message to one destination profile if specified, or to all subscribed destination profiles.

Submitting Call Home Analysis and Report Requests

You can use the **call-home request** command to submit information about your system to Cisco to receive helpful analysis and report information specific to your system. You can request a variety of reports, including security alerts, known bugs, best practices, and command references.

Note the following guidelines when manually sending Call Home analysis and report requests:

- If a **profile** *name* is specified, the request is sent to the profile. If no profile is specified, the request is sent to the Cisco TAC profile. The recipient profile does not need to be enabled for the call-home request. The profile should specify the e-mail address where the transport gateway is configured so that the request message can be forwarded to the Cisco TAC and the user can receive the reply from the Smart Call Home service.
- The **ccoid** *user-id* is the registered identifier of the Smart Call Home user. If the *user-id* is specified, the response is sent to the e-mail address of the registered user. If no *user-id* is specified, the response is sent to the contact e-mail address of the device.
- Based on the keyword specifying the type of report requested, the following information is returned:
  - **config-sanity**—Information on best practices as related to the current running configuration.
  - **bugs-list**—Known bugs in the running version and in the currently applied features.
  - **command-reference**—Reference links to all commands in the running configuration.
  - **product-advisory**—Product Security Incident Response Team (PSIRT) notices, End of Life (EOL) or End of Sales (EOS) notices, or field notices (FN) that may affect the devices in your network.

To submit a request for analysis and report information from the Cisco Output Interpreter tool, perform the following steps:

SUMMARY STEPS

1. **call-home request output-analysis** *"show-command"* [**profile** *name*] [**ccoid** *user-id*]
2. **call-home request** {**config-sanity** | **bugs-list** | **command-reference** | **product-advisory**} [**profile** *name*] [**ccoid** *user-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**call-home request output-analysis** *"show-command"* [**profile** *name*] [**ccoid** *user-id*] **Example:** `Router# call-home request output-analysis "show diag" profile TG`	Sends the output of the specified show command for analysis. The show command must be contained in quotes ("").
Step 2	**call-home request** {**config-sanity** | **bugs-list** | **command-reference** | **product-advisory**} [**profile** *name*] [**ccoid** *user-id*] **Example:** `Router# call-home request config-sanity profile TG`	Sends the output of a predetermined set of commands such as the **show running-config all**, **show version** or **show module** commands, for analysis. In addition, the **call home request product-advisory** sub-command includes all inventory alert group commands. The keyword specified after **request** specifies the type of report requested.

Example

The following example shows a request for analysis of a user-specified **show** command:

```
Router# call-home request output-analysis "show diag" profile TG
```

Manually Sending Command Output Message for One Command or a Command List

You can use the **call-home send** command to execute an IOS command or a list of IOS commands and send the command output through HTTP or e-mail protocol.

Note the following guidelines when sending the output of a command:

- The specified IOS command or list of IOS commands can be any run command, including commands for all modules. The command must be contained in quotes (“”).
- If the e-mail option is selected using the “email” keyword and an e-mail address is specified, the command output is sent to that address. If neither the e-mail nor the HTTP option is specified, the output is sent in long-text format with the specified service request number to the Cisco TAC (attach@cisco.com).
- If neither the “email” nor the “http” keyword is specified, the service request number is required for both long-text and XML message formats and is provided in the subject line of the e-mail.
- If the HTTP option is specified, the CiscoTac-1 profile destination HTTP or HTTPS URL is used as the destination. The destination e-mail address can be specified so that Smart Call Home can forward the message to the e-mail address. The user must specify either the destination e-mail address or an SR number but they can also specify both.

To execute a command and send the command output, perform the following step:

SUMMARY STEPS

1. **call-home send** {*cli command* | *cli list*} [**email** *email* **msg-format** {**long-text** | **xml**} | **http** {**destination-email-address** *email*}] [**tac-service-request** *SR#*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**call-home send** {*cli command* | *cli list*} [**email** *email* **msg-format** {**long-text** | **xml**} | **http** {**destination-email-address** *email*}] [**tac-service-request** *SR#*] **Example:** `Router# call-home send “show version;show running-config;show inventory” email support@example.com msg-format xml`	Executes the CLI or CLI list and sends output via e-mail or HTTP. • {*cli command* | *cli list*}—Specifies the IOS command or list of IOS commands (separated by ‘;’). It can be any run command, including commands for all modules. The commands must be contained in quotes (“”). • **email** *email* **msg-format {long-text | xml}**—If the **email** option is selected, the command output will be sent to the specified e-mail address in long-text or XML format with the service request number in the subject. The e-mail address, the service request number, or both must be specified. The service request number is required if the e-mail address is not specified (default is attach@cisco.com for long-text format and callhome@cisco.com for XML format). • **http** {**destination-email-address** *email*}—If the **http** option is selected, the command output will be sent to

	Command or Action	Purpose
		Smart Call Home backend server (URL specified in TAC profile) in XML format. **destination-email-address** *email* can be specified so that the backend server can forward the message to the e-mail address. The e-mail address, the service request number, or both must be specified. • **tac-service-request** *SR#*—Specifies the service request number. The service request number is required if the e-mail address is not specified.

Example

The following example shows how to send the output of a command to a user-specified e-mail address:

```
Router# call-home send "show diag" email support@example.com
```

The following example shows the command output sent in long-text format to attach@cisco.com, with the SR number specified:

```
Router# call-home send "show version; show run" tac-service-request 123456
```

The following example shows the command output sent in XML message format to callhome@cisco.com:

```
Router# call-home send "show version; show run" email callhome@cisco.com msg-format xml
```

The following example shows the command output sent in XML message format to the Cisco TAC backend server, with the SR number specified:

```
Router# call-home send "show version; show run" http tac-service-request 123456
```

The following example shows the command output sent to the Cisco TAC backend server through the HTTP protocol and forwarded to a user-specified email address:

```
Router# call-home send "show version; show run" http destination-email-address
user@company.com
```

Configuring Diagnostic Signatures

The Diagnostic Signatures feature downloads digitally signed signatures to devices. Diagnostic Signatures (DS) files are formatted files that collate knowledge of diagnostic events and provide methods to troubleshoot them without a need to upgrade the Cisco software. The aim of DS is to deliver flexible intelligence that can detect and collect troubleshooting information that can be used to resolve known problems in customers networks.

Information About Diagnostic Signatures

Diagnostic Signatures Overview

Diagnostic signatures (DS) for the Call Home system provides a flexible framework that allows the defining of new events and corresponding CLIs that can analyze these events without upgrading the Cisco software.

DSs provide the ability to define more types of events and trigger types than the standard Call Home feature supports. The DS subsystem downloads and processes files on a device as well as handles callbacks for diagnostic signature events.

The Diagnostic Signature feature downloads digitally signed signatures that are in the form of files to devices. DS files are formatted files that collate the knowledge of diagnostic events and provide methods to troubleshoot these events.

DS files contain XML data to specify the event description, and these files include CLI commands or scripts to perform required actions. These files are digitally signed by Cisco or a third party to certify their integrity, reliability, and security.

The structure of a DS file can be one of the following formats:

- Metadata-based simple signature that specifies the event type and contains other information that can be used to match the event and perform actions such as collecting information by using the CLI. The signature can also change configurations on the device as a workaround for certain bugs.
- Embedded Event Manager (EEM) Tool Command Language (Tcl) script-based signature that specifies new events in the event register line and additional action in the Tcl script.
- Combination of both the formats above.

The following basic information is contained in a DS file:

- **ID (unique number)**—Unique key that represents a DS file that can be used to search a DS.
- **Name (ShortDescription)**—Unique description of the DS file that can be used in lists for selection.
- **Description**—Long description about the signature.
- **Revision**—Version number, which increments when the DS content is updated.
- **Event & Action**—Defines the event to be detected and the action to be performed after the event happens.

Prerequisites for Diagnostic Signatures

Before you download and configure diagnostic signatures (DSs) on a device, you must ensure that the following conditions are met:

- You must assign one or more DSs to the device. For more information on how to assign DSs to devices, see Downloading Diagnostic Signatures, on page 328.
- HTTP/Secure HTTP (HTTPS) transport is required for downloading DS files. You must install the certification authority (CA) certificate to enable the authentication of the destination HTTPS server.

Note If you configure the trustpool feature, the CA certificate is not required.

Downloading Diagnostic Signatures

To download the diagnostic signature (DS) file, you require the secure HTTP (HTTPS) protocol. If you have already configured an email transport method to download files on your device, you must change your assigned profile transport method to HTTPS to download and use DS.

Cisco software uses a PKI Trustpool Management feature, which is enabled by default on devices, to create a scheme to provision, store, and manage a pool of certificates from known certification authorities (CAs). The trustpool feature installs the CA certificate automatically. The CA certificate is required for the authentication of the destination HTTPS servers.

There are two types of DS update requests to download DS files: regular and forced-download. Regular download requests DS files that were recently updated. You can trigger a regular download request either by using a periodic configuration or by initiating an on-demand CLI. The regular download update happens only when the version of the requested DS is different from the version of the DS on the device. Periodic download is only started after there is any DS assigned to the device from DS web portal. After the assignment happens, the response to the periodic inventory message from the same device will include a field to notify device to start its periodic DS download/update. In a DS update request message, the status and revision number of the DS is included such that only a DS with the latest revision number is downloaded.

Forced-download downloads a specific DS or a set of DSes. You can trigger the forced-download update request only by initiating an on-demand CLI. In a force-download update request, the latest version of the DS file is downloaded irrespective of the current DS file version on the device.

The DS file is digitally signed, and signature verification is performed on every downloaded DS file to make sure it is from a trusted source.

Diagnostic Signature Workflow

The diagnostic signature feature is enabled by default in Cisco software. The following is the workflow for using diagnostic signatures:

- Find the DS(es) you want to download and assign them to the device. This step is mandatory for regular periodic download, but not required for forced download.
- The device downloads all assigned DS(es) or a specific DS by regular periodic download or by on-demand forced download.

- The device verifies the digital signature of every single DS. If verification passes, the device stores the DS file into a non-removable disk, such as bootflash or hard disk, so that DS files can be read after the device is reloaded. On the router, the DS file is stored in the bootflash:/call home directory.
- The device continues sending periodic regular DS download requests to get the latest revision of DS and replace the older one in device.
- The device monitors the event and executes the actions defined in the DS when the event happens.

Diagnostic Signature Events and Actions

The events and actions sections are the key areas used in diagnostic signatures. The event section defines all event attributes that are used for event detection. The action section lists all actions which should be performed after the event happens, such as collecting show command outputs and sending them to Smart Call Home to parse.

Diagnostic Signature Event Detection

Event detection in a DS is defined in two ways: single event detection and multiple event detection.

Single Event Detection

In single event detection, only one event detector is defined within a DS. The event specification format is one of the following two types:

- DS event specification type: syslog, periodic, configuration, Online Insertion Removal (OIR) immediate, and call home are the supported event types, where "immediate" indicates that this type of DS does not detect any events, its actions are performed once it is downloaded, and the call-home type modifies the current CLI commands defined for existing alert-group.
- The Embedded Event Manager (EEM) specification type: supports any new EEM event detector without having to modify the Cisco software.

  Other than using EEM to detect events, a DS is triggered when a Tool Command Language (Tcl) script is used to specify event detection types.

Multiple Event Detection

Multiple event detection involves defining two or more event detectors, two ore more corresponding tracked object states, and a time period for the events to occur. The specification format for multiple event detection can include complex event correlation for tracked event detectors. For example, three event detectors (syslog, OIR, and IPSLA) are defined during the creation of a DS file. The correlation that is specified for these event detectors is that the DS will execute its action if both syslog and OIR events are triggered simultaneously, or if IPSLA is triggered alone.

Diagnostic Signature Actions

The diagnostic signature (DS) file consists of various actions that must be initiated when an event occurs. The action type indicates the kind of action that will be initiated in response to a certain event.

Variables are elements within a DS that are used to customize the files.

DS actions are categorized into the following four types:

- call-home

- command
- emailto
- script

DS action types call-home and emailto collect event data and send a message to call-home servers or to the defined email addresses. The message uses “diagnostic-signature” as its message type and DS ID as the message sub-type.

The commands defined for the DS action type initiate CLI commands that can change configuration of the device, collect show command outputs, or run any EXEC command on the device. The DS action type script executes Tcl scripts.

Diagnostic Signature Variables

Variables are referenced within a DS and are used to customize the DS file. All DS variable names have the prefix ds_ to separate them from other variables. The following are the supported DS variable types:

- System variable: variables assigned automatically by the device without any configuration changes. The Diagnostic Signatures feature supports two system variables: ds_hostname and ds_signature_id.
- Environment variable: values assigned manually by using the **environment** *variable-name variable-value* command in call-home diagnostic-signature configuration mode. Use the **show call-home diagnostic-signature** command to display the name and value of all DS environment variables. If the DS file contains unresolved environment variables, this DS will stay in pending status until the variable gets resolved.
- Prompt variable: values assigned manually by using the **call-home diagnostic-signature install** *ds-id* command in privileged EXEC mode. If you do not set this value, the status of the DS indicates pending.
- Regular expression variable: values assigned from a regular expression pattern match with predefined CLI command outputs. The value is assigned during the DS run.
- Syslog event variable: values assigned during a syslog event detection in the DS file. This variable is valid only for syslog event detection.

How to Configure Diagnostic Signatures

- Configuring the Call Home Service for Diagnostic Signatures, on page 330
- Configuring Diagnostic Signatures, on page 332

Configuring the Call Home Service for Diagnostic Signatures

Configure the Call Home Service feature to set attributes such as the contact email address where notifications related with diagnostic signatures (DS) are sent and destination HTTP/secure HTTP (HTTPS) URL to download the DS files from.

You can also create a new user profile, configure correct attributes and assign it as the DS profile. For periodic downloads, the request is sent out just following full inventory message. By changing the inventory periodic configuration, the DS periodic download also gets rescheduled.

Note The predefined CiscoTAC-1 profile is enabled as a DS profile by default and we recommend that you use it. If used, you only need to change the destination transport-method to the **http** setting.

SUMMARY STEPS

1. **configure terminal**
2. **service call-home**
3. **call-home**
4. **contact-email-addr** *email-address*
5. **mail-server** {*ipv4-addr* | *name*} **priority** *number*
6. **profile** *profile-name*
7. **destination transport-method** {**email** | **http**}
8. **destination address** {**email** *address* | **http** *url*}
9. **subscribe-to-alert-group inventory** [**periodic** {**daily** *hh:mm* | **monthly** *day hh:mm* | **weekly** *day hh:mm*}]
10. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**service call-home** **Example:** `Router(config)# service call-home`	Enables Call Home service on a device.
Step 3	**call-home** **Example:** `Router(config)# call-home`	Enters call-home configuration mode for the configuration of Call Home settings.
Step 4	**contact-email-addr** *email-address* **Example:** `Router(cfg-call-home)# contact-email-addr userid@example.com`	(Optional) Assigns an email address to be used for Call Home customer contact.
Step 5	**mail-server** {*ipv4-addr* | *name*} **priority** *number* **Example:** `Router(cfg-call-home)# mail-server 10.1.1.1 priority 4`	(Optional) Configures a Simple Mail Transfer Protocol (SMTP) email server address for Call Home. This command is only used when sending email is part of the actions defined in any DS.
Step 6	**profile** *profile-name* **Example:**	Configures a destination profile for Call Home and enters call-home profile configuration mode.

	Command or Action	Purpose
	`Router(cfg-call-home)# profile user1`	
Step 7	**destination transport-method** {**email** | **http**} **Example:** `Router(cfg-call-home-profile)# destination transport-method http`	Specifies a transport method for a destination profile in the Call Home. **Note** To configure diagnostic signatures, you must use the **http** option.
Step 8	**destination address** {**email** *address* | **http** *url*} **Example:** `Router(cfg-call-home-profile)# destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService`	Configures the address type and location to which call-home messages are sent. **Note** To configure diagnostic signatures, you must use the **http** option.
Step 9	**subscribe-to-alert-group inventory** [**periodic** {**daily** *hh:mm* | **monthly** *day hh:mm* | **weekly** *day hh:mm*}] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group inventory periodic daily 14:30`	Configures a destination profile to send messages for the Inventory alert group for Call Home. • This command is used only for the periodic downloading of DS files.
Step 10	**exit** **Example:** `Router(cfg-call-home-profile)# exit`	Exits call-home profile configuration mode and returns to call-home configuration mode.

What to do next

Set the profile configured in the previous procedure as the DS profile and configure other DS parameters.

Configuring Diagnostic Signatures

Before you begin

Configure the Call Home feature to set attributes for the Call Home profile. You can either use the default CiscoTAC-1 profile or use the newly-created user profile.

SUMMARY STEPS

1. **call-home**
2. **diagnostic-signature**
3. **profile** *ds-profile-name*
4. **environment** *ds_env-var-name* *ds-env-var-value*
5. **end**
6. **call-home diagnostic-signature** [{**deinstall** | **download**} {*ds-id* | **all**} | **install** *ds-id*]
7. **show call-home diagnostic-signature** [*ds-id* {**actions** | **events** | **prerequisite** | **prompt** | **variables** | **failure** | **statistics** | **download**}]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**call-home** **Example:** `Router(config)# call-home`	Enters call-home configuration mode for the configuration of Call Home settings.
Step 2	**diagnostic-signature** **Example:** `Router(cfg-call-home)# diagnostic-signature`	Enters call-home diagnostic signature mode.
Step 3	**profile** *ds-profile-name* **Example:** `Router(cfg-call-home-diag-sign)# profile user1`	Specifies the destination profile on a device that DS uses.
Step 4	**environment** *ds_env-var-name* *ds-env-var-value* **Example:** `Router(cfg-call-home-diag-sign)# environment ds_env1 envarval`	Sets the environment variable value for DS on a device.
Step 5	**end** **Example:** `Router(cfg-call-home-diag-sign)# end`	Exits call-home diagnostic signature mode and returns to privileged EXEC mode.
Step 6	**call-home diagnostic-signature** [{**deinstall** | **download**} {*ds-id* | **all**} | **install** *ds-id*] **Example:** `Router# call-home diagnostic-signature download 6030`	Downloads, installs, and uninstalls diagnostic signature files on a device.
Step 7	**show call-home diagnostic-signature** [*ds-id* {**actions** | **events** | **prerequisite** | **prompt** | **variables** | **failure** | **statistics** | **download**}] **Example:** `Router# show call-home diagnostic-signature actions`	Displays the call-home diagnostic signature information.

Configuration Examples for Diagnostic Signatures

The following example shows how to enable the periodic downloading request for diagnostic signature (DS) files. This configuration will send download requests to the service call-home server daily at 2:30 p.m. to check for updated DS files. The transport method is set to HTTP.

```
Router> enable
Router# configure terminal
Router(config)# service call-home
Router(config)# call-home
Router(cfg-call-home)# contact-email-addr userid@example.com
Router(cfg-call-home)# mail-server 10.1.1.1 priority 4
Router(cfg-call-home)# profile user-1
```

```
Router(cfg-call-home-profile)# destination transport-method http
Router(cfg-call-home-profile)# destination address http
https://tools.cisco.com/its/service/oddce/services/DDCEService
Router(cfg-call-home-profile)# subscribe-to-alert-group inventory periodic daily 14:30
Router(cfg-call-home-profile)# exit
Router(cfg-call-home)# diagnostic-signature
Router(cfg-call-home-diag-sign)# profile user1
Router(cfg-call-home-diag-sign)# environment ds_env1 envarval
Router(cfg-call-home-diag-sign)# end
```

The following is sample output from the **show call-home diagnostic-signature** command for the configuration displayed above:

```
outer# show call-home diagnostic-signature

Current diagnostic-signature settings:
Diagnostic-signature: enabled
Profile: user1 (status: ACTIVE)
Environment variable:
ds_env1: abc
Downloaded DSes:
DS ID    DS Name                          Revision Status     Last Update (GMT+00:00)
-------- -------------------------------- -------- ---------- -------------------
6015     CronInterval                     1.0      registered 2013-01-16 04:49:52
6030     ActCH                            1.0      registered 2013-01-16 06:10:22
6032     MultiEvents                      1.0      registered 2013-01-16 06:10:37
6033     PureTCL                          1.0      registered 2013-01-16 06:11:48
```

Displaying Call Home Configuration Information

You can use variations of the **show call-home** command to display Call Home configuration information.

To display the configured Call Home information, perform the following:

SUMMARY STEPS

1. **show call-home**
2. **show call-home detail**
3. **show call-home alert-group**
4. **show call-home mail-server status**
5. **show call-home profile** {**all** | *name*}
6. **show call-home statistics** [**detail** | **profile** *profile_name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show call-home** **Example:** `Router# show call-home`	Displays the Call Home configuration in summary.
Step 2	**show call-home detail** **Example:**	Displays the Call Home configuration in detail.

	Command or Action	Purpose
	`Router# show call-home detail`	
Step 3	**show call-home alert-group** **Example:** `Router# show call-home alert-group`	Displays the available alert groups and their status.
Step 4	**show call-home mail-server status** **Example:** `Router# show call-home mail-server status`	Checks and displays the availability of the configured e-mail server(s).
Step 5	**show call-home profile** {**all** | *name*} **Example:** `Router# show call-home profile all`	Displays the configuration of the specified destination profile. Use the **all** keyword to display the configuration of all destination profiles.
Step 6	**show call-home statistics** [**detail** | **profile** *profile_name*] **Example:** `Router# show call-home statistics`	Displays the statistics of Call Home events.

Examples

Call Home Information in Summary

Call Home Information in Detail

Available Call Home Alert Groups

E-Mail Server Status Information

Information for All Destination Profiles

Information for a User-Defined Destination Profile

Call Home Statistics

The following examples show the sample output when using different options of the **show call-home** command.

```
Router# show call-home
Current call home settings:
    call home feature : enable
    call home message's from address: router@example.com
    call home message's reply-to address: support@example.com

    vrf for call-home messages: Not yet set up

    contact person's email address: technical@example.com
```

```
    contact person's phone number: +1-408-555-1234
    street address: 1234 Picaboo Street, Any city, Any state, 12345
    customer ID: ExampleCorp
    contract ID: X123456789
    site ID: SantaClara

    source ip address: Not yet set up
    source interface: GigabitEthernet0/0
    Mail-server[1]: Address: 192.0.2.2 Priority: 1
    Mail-server[2]: Address: 203.0.113.1 Priority: 2
    http proxy: 192.0.2.1:80

    aaa-authorization: disable
    aaa-authorization username: callhome (default)
    data-privacy: normal
    syslog throttling: enable

    Rate-limit: 20 message(s) per minute

    Snapshot command[0]: show version
    Snapshot command[1]: show clock

Available alert groups:
    Keyword                  State   Description
    ------------------------ ------- -------------------------------
    configuration            Enable  configuration info
    crash                    Enable  crash and traceback info
    environment              Enable  environmental info
    inventory                Enable  inventory info
    snapshot                 Enable  snapshot info
    syslog                   Enable  syslog info

Profiles:
    Profile Name: campus-noc
    Profile Name: CiscoTAC-1
Router#

Router# show call-home detail
Current call home settings:
    call home feature : enable
    call home message's from address: router@example.com
    call home message's reply-to address: support@example.com

    vrf for call-home messages: Not yet set up

    contact person's email address: technical@example.com

    contact person's phone number: +1-408-555-1234
    street address: 1234 Picaboo Street, Any city, Any state, 12345
    customer ID: ExampleCorp
    contract ID: X123456789
    site ID: SantaClara

    source ip address: Not yet set up
    source interface: GigabitEthernet0/0
    Mail-server[1]: Address: 192.0.2.2 Priority: 1
    Mail-server[2]: Address: 203.0.113.1 Priority: 2
    http proxy: 192.0.2.1:80

    aaa-authorization: disable
    aaa-authorization username: callhome (default)
    data-privacy: normal
    syslog throttling: enable

    Rate-limit: 20 message(s) per minute
```

```
    Snapshot command[0]: show version
    Snapshot command[1]: show clock

Available alert groups:
    Keyword                  State   Description
    ------------------------ ------- -------------------------------
    configuration            Enable  configuration info
    crash                    Enable  crash and traceback info
    environment              Enable  environmental info
    inventory                Enable  inventory info
    snapshot                 Enable  snapshot info
    syslog                   Enable  syslog info

Profiles:

Profile Name: campus-noc
    Profile status: ACTIVE
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): noc@example.com
    HTTP  address(es): Not yet set up

    Alert-group               Severity
    ------------------------  ------------
    configuration             normal
    crash                     normal
    environment               debug
    inventory                 normal

    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*              debug

Profile Name: CiscoTAC-1
    Profile status: INACTIVE
    Profile mode: Full Reporting
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): callhome@cisco.com
    HTTP  address(es): https://tools.cisco.com/its/service/oddce/services/DDCEService

    Periodic configuration info message is scheduled every 14 day of the month at 11:12

    Periodic inventory info message is scheduled every 14 day of the month at 10:57

    Alert-group               Severity
    ------------------------  ------------
    crash                     normal
    environment               minor

    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*              debug
Router#

Router# show call-home alert-group
Available alert groups:
    Keyword                  State   Description
    ------------------------ ------- -------------------------------
    configuration            Enable  configuration info
    crash                    Enable  crash and traceback info
    environment              Enable  environmental info
```

```
    inventory                Enable  inventory info
    snapshot                 Enable  snapshot info
    syslog                   Enable  syslog info
Router#

Router# show call-home mail-server status
Please wait. Checking for mail server status ...

    Mail-server[1]: Address: 192.0.2.2 Priority: 1 [Not Available]
    Mail-server[2]: Address: 203.0.113.1 Priority: 2 [Available]
Router#

Router# show call-home profile all

Profile Name: campus-noc
    Profile status: ACTIVE
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): noc@example.com
    HTTP  address(es): Not yet set up

    Alert-group               Severity
    ------------------------  ------------
    configuration             normal
    crash                     normal
    environment               debug
    inventory                 normal

    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*             debug

Profile Name: CiscoTAC-1
    Profile status: INACTIVE
    Profile mode: Full Reporting
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): callhome@cisco.com
    HTTP  address(es): https://tools.cisco.com/its/service/oddce/services/DDCEService

    Periodic configuration info message is scheduled every 14 day of the month at 11:12

    Periodic inventory info message is scheduled every 14 day of the month at 10:57

    Alert-group               Severity
    ------------------------  ------------
    crash                     normal
    environment               minor

    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*             debug
Router#

Router# show call-home profile campus-noc
Profile Name: campus-noc
    Profile status: ACTIVE
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): noc@example.com
    HTTP  address(es): Not yet set up
```

```
    Alert-group               Severity
    ------------------------  ------------
    configuration             normal
    crash                     normal
    environment               debug
    inventory                 normal

    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*             debug

Router#

Router# show call-home statistics
Message Types    Total                Email                HTTP
-------------   -------------------- -------------------- ------------------
Total Success   3                    3                    0
    Config      3                    3                    0
    Crash       0                    0                    0
    Environment 0                    0                    0
    Inventory   0                    0                    0
    Snapshot    0                    0                    0
    SysLog      0                    0                    0
    Test        0                    0                    0
    Request     0                    0                    0
    Send-CLI    0                    0                    0

Total In-Queue  0                    0                    0
    Config      0                    0                    0
    Crash       0                    0                    0
    Environment 0                    0                    0
    Inventory   0                    0                    0
    Snapshot    0                    0                    0
    SysLog      0                    0                    0
    Test        0                    0                    0
    Request     0                    0                    0
    Send-CLI    0                    0                    0

Total Failed    0                    0                    0
    Config      0                    0                    0
    Crash       0                    0                    0
    Environment 0                    0                    0
    Inventory   0                    0                    0
    Snapshot    0                    0                    0
    SysLog      0                    0                    0
    Test        0                    0                    0
    Request     0                    0                    0
    Send-CLI    0                    0                    0

Total Ratelimit
      -dropped  0                    0                    0
    Config      0                    0                    0
    Crash       0                    0                    0
    Environment 0                    0                    0
    Inventory   0                    0                    0
    Snapshot    0                    0                    0
    SysLog      0                    0                    0
    Test        0                    0                    0
    Request     0                    0                    0
    Send-CLI    0                    0                    0

Last call-home message sent time: 2011-09-26 23:26:50 GMT-08:00
Router#
```

Default Call Home Settings

The following table lists the default Call Home settings.

Table 33: Default Call Home Settings

Parameters	Default
Call Home feature status	Disabled
User-defined profile status	Active
Predefined Cisco TAC profile status	Inactive
Transport method	E-mail
Message format type	XML
Destination message size for a message sent in long text, short text, or XML format	3,145,728
Alert group status	Enabled
Call Home message severity threshold	Debug
Message rate limit for messages per minute	20
AAA Authorization	Disabled
Call Home syslog message throttling	Enabled
Data privacy level	Normal

Alert Group Trigger Events and Commands

Call Home trigger events are grouped into alert groups, with each alert group assigned commands to execute when an event occurs. The command output is included in the transmitted message. The following table lists the trigger events included in each alert group, including the severity level of each event and the executed commands for the alert group.

Table 34: Call Home Alert Groups, Events, and Actions

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
Crash	SYSTEM_ CRASH	–	–	Events related to software crash. The following commands are executed: **show version** **show logging** **show region** **show inventory** **show stack** **crashinfo file** (this command shows the contents of the crashinfo file)
–	TRACEBACK	–	–	Detects software traceback events. The following commands are executed: **show version** **show logging** **show region** **show stack**

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
Configuration	–	–	–	User-generated request for configuration or configuration change event. The following commands are executed: **show platform** **show inventory** **show running-config all** **show startup-config** **show version**
Environmental	–	–	–	Events related to power, fan, and environment sensing elements such as temperature alarms. The following commands are executed: **show environment** **show inventory** **show platform** **show logging**
–	–	SHUT	0	Environmental Monitor initiated shutdown.
–	–	ENVCRIT	2	Temperature or voltage measurement exceeded critical threshold.
–	–	BLOWER	3	Required number of fan trays is not present.

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
–	–	ENVWARN	4	Temperature or voltage measurement exceeded warning threshold.
–	–	RPSFAIL	4	Power supply may have a failed channel.
–	ENVM	PSCHANGE	6	Power supply name change.
–	–	PSLEV	6	Power supply state change.
–	–	PSOK	6	Power supply now appears to be working correctly.

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
Inventory	–	–	–	

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
				Inventory status should be provided whenever a unit is cold-booted or when FRUs are inserted or removed. This is considered a noncritical event, and the information is used for status and entitlement. Commands executed for all Inventory messages sent in anonymous mode and for Delta Inventory message sent in full registration mode: **show diag all eeprom detail** **show version** **show inventory oid** **show platform** Commands executed for Full Inventory message sent in full registration mode: **show platform** **show diag all eeprom detail** **show version** **show inventory oid** **show bootflash: all** **show data-corruption** **show interfaces** **show file systems** **show memory statistics**

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
				show process memory **show process cpu** **show process cpu history** **show license udi** **show license detail** **show buffers**
–	HARDWARE_ REMOVAL	REMCARD	6	Card removed from slot %d, interfaces disabled.
–	HARDWARE_ INSERTION	INSCARD	6	Card inserted in slot %d, interfaces administratively shut down.
Syslog	–	–	–	Event logged to syslog. The following commands are executed: **show inventory** **show logging**
–	SYSLOG	LOG_EMERG	0	System is unusable.
–	SYSLOG	LOG_ALERT	1	Action must be taken immediately.
–	SYSLOG	LOG_CRIT	2	Critical conditions.
–	SYSLOG	LOG_ERR	3	Error conditions.
–	SYSLOG	LOG_WARNING	4	Warning conditions.
–	SYSLOG	LOG_NOTICE	5	Normal but signification condition.
–	SYSLOG	LOG_INFO	6	Informational.
–	SYSLOG	LOG_DEBUG	7	Debug-level messages.

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
Test	–	TEST	–	User-generated test message. The following commands are executed: **show platform** **show inventory** **show version**

Note Cisco ISR 4321 does not display the serial numbers of power supply and fan tray with the **show inventory** command.

Message Contents

This section consists of tables which list the content formats of alert group messages.

This section also includes the following subsections that provide sample messages:

- Sample Syslog Alert Notification in Long-Text Format, on page 352
- Sample Syslog Alert Notification in XML Format, on page 354

The following table lists the content fields of a short text message.

Table 35: Format for a Short Text Message

Data Item	Description
Device identification	Configured device name
Date/time stamp	Time stamp of the triggering event
Error isolation message	Plain English description of triggering event
Alarm urgency level	Error level such as that applied to a system message

The following table shows the content fields that are common to all long text and XML messages. The fields specific to a particular alert group message are inserted at a point between the common fields. The insertion point is identified in the table.

Table 36: Common Fields for All Long Text and XML Messages

Data Item (Plain Text and XML)	Description (Plain Text and XML)	Call-Home Message Tag (XML Only)
Time stamp	Date and time stamp of event in ISO time notation: *YYYY-MM-DD HH:MM:SS GMT+HH:MM*.	CallHome/EventTime
Message name	Name of message. Specific event names are listed in the Alert Group Trigger Events and Commands, on page 340.	For short text message only
Message type	Specifically “Call Home”.	CallHome/Event/Type
Message subtype	Specific type of message: full, delta, test	CallHome/Event/SubType
Message group	Specifically “reactive”. Optional because default is “reactive”.	For long-text message only
Severity level	Severity level of message (see Message Severity Threshold, on page 315).	Body/Block/Severity
Source ID	Product type for routing through the workflow engine. This is typically the product family name.	For long-text message only

Data Item (Plain Text and XML)	Description (Plain Text and XML)	Call-Home Message Tag (XML Only)
Device ID	Unique device identifier (UDI) for end device generating message. This field should be empty if the message is nonspecific to a fabric switch. The format is *type@Sid@serial*. • *type* is the product model number from backplane IDPROM. • @ is a separator character. • *Sid* is C, identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: CISCO3845@C@12345678 **Note** For the following platforms, the UDI is the Printed Circuit Board number (PCB), and not the chassis Serial Number (SN): • ISR 4221 • ISR 4321 • ISR 4331 • ISR 4351 • ISR 4431 • ISR 4451	CallHome/CustomerData/ContractData/DeviceId
Customer ID	Optional user-configurable field used for contract information or other ID by any support service.	CallHome/CustomerData/ContractData/CustomerId
Contract ID	Optional user-configurable field used for contract information or other ID by any support service.	CallHome/CustomerData/ContractData/CustomerId
Site ID	Optional user-configurable field used for Cisco-supplied site ID or other data meaningful to alternate support service.	CallHome/CustomerData/ContractData/CustomerId

Data Item (Plain Text and XML)	Description (Plain Text and XML)	Call-Home Message Tag (XML Only)
Server ID	If the message is generated from the fabric switch, this is the unique device identifier (UDI) of the switch. • *type* is the product model number from backplane IDPROM. • @ is a separator character. • *Sid* is C, identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: CISCO3845@C@12345678	For long text message only.
Message description	Short text describing the error.	CallHome/MessageDescription
Device name	Node that experienced the event. This is the host name of the device.	CallHome/CustomerData/ SystemInfo/NameName
Contact name	Name of person to contact for issues associated with the node experiencing the event.	CallHome/CustomerData/ SystemInfo/Contact
Contact e-mail	E-mail address of person identified as contact for this unit.	CallHome/CustomerData/ SystemInfo/ContactEmail
Contact phone number	Phone number of the person identified as the contact for this unit.	CallHome/CustomerData/ SystemInfo/ContactPhoneNumber
Street address	Optional field containing street address for RMA part shipments associated with this unit.	CallHome/CustomerData/ SystemInfo/StreetAddress
Model name	Model name of the router. This is the “specific model as part of a product family name.	CallHome/Device/Cisco_Chassis/Model
Serial number	Chassis serial number of the unit.	CallHome/Device/Cisco_Chassis/ SerialNumber
Chassis part number	Top assembly number of the chassis.	CallHome/Device/ Cisco_Chassis/AdditionalInformation/ AD@name=“PartNumber”

Data Item (Plain Text and XML)	Description (Plain Text and XML)	Call-Home Message Tag (XML Only)
System object ID	System Object ID that uniquely identifies the system.	CallHome/Device/ Cisco_Chassis/AdditionalInformation/ AD@name="sysObjectID"
System description	System description for the managed element.	CallHome/Device/ Cisco_Chassis/AdditionalInformation/ AD@name="sysDescr"

The following table shows the inserted fields specific to a particular alert group message.

Note The following fields may be repeated if multiple commands are executed for this alert group.

Table 37: Inserted Fields Specific to a Particular Alert Group Message

Command output name	Exact name of the issued command.	/aml/Attachments/Attachment/Name
Attachment type	Attachment type. Usually "inline".	/aml/Attachments/Attachment@type
MIME type	Normally "text" or "plain" or encoding type.	/aml/Attachments/Attachment/ Data@encoding
Command output text	Output of command automatically executed (see Alert Group Trigger Events and Commands, on page 340).	/mml/attachments/attachment/atdata

The following table shows the inserted content fields for reactive messages (system failures that require a TAC case) and proactive messages (issues that might result in degraded system performance).

Table 38: Inserted Fields for a Reactive or Proactive Event Message

Data Item (Plain Text and XML)	Description (Plain Text and XML)	Call-Home Message Tag (XML Only)
Chassis hardware version	Hardware version of chassis	CallHome/Device/Cisco_Chassis/ HardwareVersion
Supervisor module software version	Top-level software version	CallHome/Device/Cisco_Chassis/ AdditionalInformation/AD@name= "SoftwareVersion"
Affected FRU name	Name of the affected FRU generating the event message	CallHome/Device/Cisco_Chassis/ Cisco_Card/Model
Affected FRU serial number	Serial number of affected FRU	CallHome/Device/Cisco_Chassis/ Cisco_Card/SerialNumber
Affected FRU part number	Part number of affected FRU	CallHome/Device/Cisco_Chassis/ Cisco_Card/PartNumber

Data Item (Plain Text and XML)	Description (Plain Text and XML)	Call-Home Message Tag (XML Only)
FRU slot	Slot number of FRU generating the event message	CallHome/Device/Cisco_Chassis/Cisco_Card/LocationWithinContainer
FRU hardware version	Hardware version of affected FRU	CallHome/Device/Cisco_Chassis/Cisco_Card/HardwareVersion
FRU software version	Software version(s) running on affected FRU	CallHome/Device/Cisco_Chassis/Cisco_Card/SoftwareIdentity/VersionString

The following table shows the inserted content fields for an inventory message.

Table 39: Inserted Fields for an Inventory Event Message

Data Item (Plain Text and XML)	Description (Plain Text and XML)	Call-Home Message Tag (XML Only)
Chassis hardware version	Hardware version of chassis	CallHome/Device/Cisco_Chassis/HardwareVersion
Supervisor module software version	Top-level software version	CallHome/Device/Cisco_Chassis/AdditionalInformation/AD@name="SoftwareVersion"
FRU name	Name of the affected FRU generating the event message	CallHome/Device/Cisco_Chassis/Cisco_Card/Model
FRU s/n	Serial number of FRU	CallHome/Device/Cisco_Chassis/Cisco_Card/SerialNumber
FRU part number	Part number of FRU	CallHome/Device/Cisco_Chassis/Cisco_Card/PartNumber
FRU slot	Slot number of FRU	CallHome/Device/Cisco_Chassis/Cisco_Card/LocationWithinContainer
FRU hardware version	Hardware version of FRU	CallHome/Device/Cisco_Chassis/CiscoCard/HardwareVersion
FRU software version	Software version(s) running on FRU	CallHome/Device/Cisco_Chassis/Cisco_Card/SoftwareIdentity/VersionString

Sample Syslog Alert Notification in Long-Text Format

The following example shows a Syslog alert notification in long-text format:

```
TimeStamp : 2014-08-13 21:41 GMT+00:00
Message Name : syslog
Message Type : Call Home
Message Group : reactive
Severity Level : 2
Source ID : ISR 4400
Device ID : ISR4451-X/K9@C@FTX1830AKF9
Customer ID :
Contract ID :
```

```
Site ID :
Server ID : ISR4451-X/K9@C@FTX1830AKF9
Event Description : *Aug 13 21:41:35.835: %CLEAR-5-COUNTERS: Clear counter on all interfaces
 by console
System Name : Router
Contact Email : admin@yourdomain.com
Contact Phone :
Street Address :
Affected Chassis : ISR4451-X/K9
Affected Chassis Serial Number : FTX1830AKF9
Affected Chassis Part No : 800-36894-03
Affected Chassis Hardware Version : 1.0
Supervisor Software Version : 15.4(20140812:034256)
Command Output Name : show logging
Attachment Type : command output
MIME Type : text/plain
Command Output Text : show logging
Syslog logging: enabled (0 messages dropped, 4 messages rate-limited, 0 flushes, 0 overruns,
 xml disabled, filtering disabled)

No Active Message Discriminator.



No Inactive Message Discriminator.


    Console logging: level debugging, 71 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 73 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled

No active filter modules.

    Trap logging: level informational, 70 message lines logged
        Logging Source-Interface:       VRF Name:

Log Buffer (4096 bytes):

*Aug 13 21:38:04.994: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
*Aug 13 21:40:55.706: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
*Aug 13 21:41:27.042: %SYS-5-CONFIG_I: Configured from console by console
Router#
Command Output Name : show inventory
Attachment Type : command output
MIME Type : text/plain
Command Output Text : show inventory
NAME: "Chassis", DESCR: "Cisco ISR4451 Chassis"
PID: ISR4451-X/K9      , VID: V03, SN: FTX1830AKF9

NAME: "Power Supply Module 0", DESCR: "450W AC Power Supply for Cisco ISR4450, ISR4350"
PID: PWR-4450-AC       , VID: V01, SN: DCA1822X0G4

NAME: "Fan Tray", DESCR: "Cisco ISR4450, ISR4350 Fan Assembly"
PID: ACS-4450-FANASSY  , VID:    , SN:

NAME: "module 0", DESCR: "Cisco ISR4451 Built-In NIM controller"
PID: ISR4451-X/K9      , VID:    , SN:
```

```
NAME: "NIM subslot 0/0", DESCR: "Front Panel 4 ports Gigabitethernet Module"
PID: ISR4451-X-4x1GE   , VID: V01, SN: JAB092709EL

NAME: "module 1", DESCR: "Cisco ISR4451 Built-In SM controller"
PID: ISR4451-X/K9      , VID:    , SN:

NAME: "module 2", DESCR: "Cisco ISR4451 Built-In SM controller"
PID: ISR4451-X/K9      , VID:    , SN:

NAME: "module R0", DESCR: "Cisco ISR4451 Route Processor"
PID: ISR4451-X/K9      , VID: V03, SN: FOC18271QLX

NAME: "module F0", DESCR: "Cisco ISR4451 Forwarding Processor"
PID: ISR4451-X/K9      , VID:    , SN:


Router#
```

Sample Syslog Alert Notification in XML Format

The following example shows a Syslog alert notification in XML format:


```
<?xml version="1.0" encoding="UTF-8"?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.cisco.com/2004/01/aml-session"
soap-env:mustUnderstand="true"
soap-env:role="http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.cisco.com/neddce/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.cisco.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.cisco.com/appliance/uri</aml-session:From>
<aml-session:MessageId>M4:FTX1830AKF9:53EBDBDA</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.cisco.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.cisco.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2014-08-13 21:42:50 GMT+00:00</aml-block:CreationDate>
<aml-block:Builder>
<aml-block:Name>ISR 4400</aml-block:Name>
<aml-block:Version>2.0</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>G5:FTX1830AKF9:53EBDBDA</aml-block:GroupId>
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>2</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:CallHome xmlns:ch="http://www.cisco.com/2005/05/callhome" version="1.0">
<ch:EventTime>2014-08-13 21:42:49 GMT+00:00</ch:EventTime>
<ch:MessageDescription>*Aug 13 21:42:49.406: %CLEAR-5-COUNTERS: Clear counter on all
interfaces by console</ch:MessageDescription>
<ch:Event>
<ch:Type>syslog</ch:Type>
<ch:SubType></ch:SubType>
<ch:Brand>Cisco Systems</ch:Brand>
```

```
<ch:Series>ISR XE Series Routers</ch:Series>
</ch:Event>
<ch:CustomerData>
<ch:UserData>
<ch:Email>admin@yourdomain.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:CustomerId></ch:CustomerId>
<ch:SiteId></ch:SiteId>
<ch:ContractId></ch:ContractId>
<ch:DeviceId>ISR4451-X/K9@C@FTX1830AKF9</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>Router</ch:Name>
<ch:Contact></ch:Contact>
<ch:ContactEmail>admin@yourdomain.com</ch:ContactEmail>
<ch:ContactPhoneNumber></ch:ContactPhoneNumber>
<ch:StreetAddress></ch:StreetAddress>
</ch:SystemInfo>
<ch:CCOID></ch:CCOID>
</ch:CustomerData>
<ch:Device>
<rme:Chassis xmlns:rme="http://www.cisco.com/rme/4.0">
<rme:Model>ISR4451-X/K9</rme:Model>
<rme:HardwareVersion>1.0</rme:HardwareVersion>
<rme:SerialNumber>FTX1830AKF9</rme:SerialNumber>
<rme:AdditionalInformation>
<rme:AD name="PartNumber" value="800-36894-03" />
<rme:AD name="SoftwareVersion" value="15.4(20140812:034256)" />
<rme:AD name="SystemObjectId" value="1.3.6.1.4.1.9.1.1707" />
<rme:AD name="SystemDescription" value="Cisco IOS Software, ISR Software
(X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version 15.4(20140812:034256)
[v154_3_s_xe313_throttle-BLD-BLD_V154_3_S_XE313_THROTTLE_LATEST_20140812_020034-ios 150]
Copyright (c) 1986-2014 by Cisco Systems, Inc.
Compiled Tue 12-Aug-14 00:13 by mcpre" />
<rme:AD name="ServiceNumber" value="" />
<rme:AD name="ForwardAddress" value="" />
</rme:AdditionalInformation>
</rme:Chassis>
</ch:Device>
</ch:CallHome>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging</aml-block:Name>
<aml-block:Data encoding="plain">
<![CDATA[show logging
Syslog logging: enabled (0 messages dropped, 4 messages rate-limited, 0 flushes, 0 overruns,
 xml disabled, filtering disabled)

No Active Message Discriminator.

No Inactive Message Discriminator.

    Console logging: level debugging, 75 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 77 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled
```

```
No active filter modules.

    Trap logging: level informational, 74 message lines logged
        Logging Source-Interface:       VRF Name:

Log Buffer (4096 bytes):

*Aug 13 21:42:20.187: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
*Aug 13 21:42:23.364: %SYS-5-CONFIG_I: Configured from console by console
Router#]]></aml-block:Data>
</aml-block:Attachment>
<aml-block:Attachment type="inline">
<aml-block:Name>show inventory</aml-block:Name>
<aml-block:Data encoding="plain">
<![CDATA[show inventory
NAME: "Chassis", DESCR: "Cisco ISR4451 Chassis"
PID: ISR4451-X/K9      , VID: V03, SN: FTX1830AKF9

NAME: "Power Supply Module 0", DESCR: "450W AC Power Supply for Cisco ISR4450, ISR4350"
PID: PWR-4450-AC       , VID: V01, SN: DCA1822X0G4

NAME: "Fan Tray", DESCR: "Cisco ISR4450, ISR4350 Fan Assembly"
PID: ACS-4450-FANASSY  , VID:    , SN:

NAME: "module 0", DESCR: "Cisco ISR4451 Built-In NIM controller"
PID: ISR4451-X/K9      , VID:    , SN:

NAME: "NIM subslot 0/0", DESCR: "Front Panel 4 ports Gigabitethernet Module"
PID: ISR4451-X-4x1GE   , VID: V01, SN: JAB092709EL

NAME: "module 1", DESCR: "Cisco ISR4451 Built-In SM controller"
PID: ISR4451-X/K9      , VID:    , SN:

NAME: "module 2", DESCR: "Cisco ISR4451 Built-In SM controller"
PID: ISR4451-X/K9      , VID:    , SN:

NAME: "module R0", DESCR: "Cisco ISR4451 Route Processor"
PID: ISR4451-X/K9      , VID: V03, SN: FOC18271QLX

NAME: "module F0", DESCR: "Cisco ISR4451 Forwarding Processor"
PID: ISR4451-X/K9      , VID:    , SN:

Router#]]></aml-block:Data>
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>
```

Additional References

The following sections provide references related to the Call Home feature.

Related Documents

Document Title	Description
Smart Call Home User Guide	Explains how the Smart Call Home service offers web-based access to important information on select Cisco devices and offers higher network availability, and increased operational efficiency by providing proactive diagnostics and real-time alerts.

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

CHAPTER 22

Configuring Bridge Domain Interfaces

The Cisco 4000 Series ISR devices support the bridge domain interface (BDI) feature for packaging Layer 2 Ethernet segments into Layer 3 IP address.

Restrictions for Bridge Domain Interfaces

The following are the restrictions pertaining to bridge domain interfaces:

- Only 4096 bridge domain interfaces are supported per system
- For a bridge domain interface, the maximum transmission unit (MTU) size can be configured between 1500 and 9216 bytes.
- Bridge domain interfaces support only the following features:
  - IPv4 Multicast
  - QoS marking and policing. Shaping and queuing are not supported
  - IPv4 VRF
  - IPv6 unicast forwarding
  - Dynamic routing such as BGP, OSPF, EIGRP, RIP, IS-IS, and STATIC
  - Hot Standby Router Protocol (HSRP) from IOS XE 3.8.0 onwards.
  - Virtual Router Redundancy Protocol (VRRP) from IOS XE 3.8.0 onwards.
  - Flexible NetFlow

Note Flexible NetFlow is supported from Cisco IOS XE 17.7.1a and later releases.

- Bridge domain interfaces do not support the following features:
  - PPP over Ethernet (PPPoE)
  - Bidirectional Forwarding Detection (BFD) protocol
  - QoS
  - Network-Based Application Recognition (NBAR) or Advanced Video Coding (AVC)

Information About Bridge Domain Interface

Bridge domain interface is a logical interface that allows bidirectional flow of traffic between a Layer 2 bridged network and a Layer 3 routed network traffic. Bridge domain interfaces are identified by the same index as the bridge domain. Each bridge domain represents a Layer 2 broadcast domain. Only one bridge domain interface can be associated with a bridge domain.

Bridge domain interface supports the following features:

- IP termination
- Layer 3 VPN termination
- Address Resolution Protocol (ARP), G-ARP, and P-ARP handling
- MAC address assignment

Prior to configuring a bridge domain interface, you must understand the following concepts:

- Ethernet Virtual Circuit Overview
- Bridge Domain Interface Encapsulation
- Assigning a MAC Address
- Support for IP Protocols
- Support for IP Forwarding
- Packet Forwarding
- Bridge Domain Interface Statistics

Ethernet Virtual Circuit Overview

An Ethernet Virtual Circuit (EVC) is an end-to-end representation of a single instance of a Layer 2 service that is offered by a provider. It embodies the different parameters on which the service is being offered. In the Cisco EVC Framework, the bridge domains are made up of one or more Layer 2 interfaces known as service instances. A service instance is the instantiation of an EVC on a given port on a given router. Service instance is associated with a bridge domain based on the configuration.

An incoming frame can be classified as service instance based on the following criteria:

- Single 802.1Q VLAN tag, priority-tagged, or 802.1ad VLAN tag
- Both QinQ (inner and outer) VLAN tags, or both 802.1ad S-VLAN and C-VLAN tags

- Outer 802.1p CoS bits, inner 802.1p CoS bits, or both
- Payload Ethernet type (five choices are supported: IPv4, IPv6, PPPoE-all, PPoE-discovery, and PPPoE-session)

Service instance also supports alternative mapping criteria:

- Untagged—Mapping to all the frames lacking a 802.1Q or 802.1ad header
- Default—Mapping to all the frames

For more information on the EVC architecture, see the section *Configuring Ethernet Virtual Connections on the Cisco ASR 1000 Router* in the Carrier Ethernet Configuration Guide .

Bridge Domain Interface Encapsulation

Security Group classification includes both Source and Destination Group, which is specified by source SGT and DGT. SGT Based PBR feature provides the PBR route-map match clause for SGT/DGT based packet classification. SGT Based PBR feature supports configuration of unlimited number of tags, but it is recommended to configure the tags based on memory available in the platform.

An EVC provides the ability to employ different encapsulations on each Ethernet flow point (EFP) present in a bridge domain. A BDI egress point may not be aware of the encapsulation of an egress packet because the packet may have egressed from one or more EFPs with different encapsulations.

In a bridge domain, if all the EFPs have different encapsulations, the BDI must be untagged (using the no 802.1Q tag). Encapsulate all the traffic in the bridge domain (popped or pushed) at the EFPs. Configure rewrite at each EFP to enable encapsulation of the traffic on the bridge domain.

In a bridge domain, if all the EFPs have the same encapsulation, configure the encapsulations on the BDI using the encapsulation command. Enabling encapsulation at the BDI ensures effective pushing or popping of tags, thereby eliminating the need for configuring the rewrite command at the EFPs. For more information on configuring the encapsulations on the BDI, see the How to Configure a Bridge Domain Interface.

Assigning a MAC Address

All the bridge domain interfaces on the Cisco 4000 Series ISR chassis share a common MAC address. The first bridge domain interface on a bridge domain is allocated a MAC address. Thereafter, the same MAC address is assigned to all the bridge domain interfaces that are created in that bridge domain.

Note You can configure a static MAC address on a bridge domain interface using the **mac-address** command.

Support for IP Protocols

Bridge domain interfaces enable the Cisco 4000 Series ISR devices to act as a Layer 3 endpoint on the Layer 2 bridge domain for the following IP-related protocols:

- ARP
- DHCP

- HTTP
- ICMP
- NTP
- RARP
- SNMP
- TCP
- Telnet
- TFTP
- UDP

Support for IP Forwarding

Bridge domain interface supports the following IP forwarding features:

- IPv4 input and output access control lists (ACL)
- IPv4 input and output QoS policies. The operations supported for the input and output service policies on a bridge domain interface are:
    - Classification
    - Marking
    - Policing
- IPv4 L3 VRFs

Packet Forwarding

A bridge domain interface provides bridging and forwarding services between the Layer 2 and Layer 3 network infrastructure.

Layer 2 to Layer 3

During a packet flow from a Layer 2 network to a Layer 3 network, if the destination MAC address of the incoming packet matches the bridge domain interface MAC address, or if the destination MAC address is a multicast address, the packet or a copy of the packet is forwarded to the bridge domain interface.

Note MAC address learning cannot not be performed on the bridge domain interface.

Layer 3 to Layer 2

When a packet arrives at a Layer 3 physical interface of a router, a route lookup action is performed. If route lookup points to a bridge domain interface, then the bridge domain interface adds the layer 2 encapsulation and forwards the frame to the corresponding bridge domain. The byte counters are updated.

During a Layer 2 lookup on a bridge domain to which the bridge domain interface belongs, the bridge domain forwards the packets to the correct service instance based on the destination MAC address.

Link States of a Bridge Domain and a Bridge Domain Interface

Bridge domain interface acts as a routable IOS interface on Layer 3 and as a port on a bridge domain. Both bridge domain interfaces and bridge domains operate with individual administrative states.

Shutting down a bridge domain interface stops the Layer 3 data service, but does not override or impact the state of the associated bridge domain.

Shutting down a bridge domain stops Layer 2 forwarding across all the associated members including service instances and bridge domain interfaces. The associated service instances influence the operational state of a bridge domain. Bridge domain interface cannot be operational unless one of the associated service instances is up.

Note Because a bridge domain interface is an internal interface, the operational state of bridge domain interface does not affect the bridge domain operational state.

BDI Initial State

The initial administrative state of a BDI depends on how the BDI is created. When you create a BDI at boot time in the startup configuration, the default administrative state for the BDI is up. It will remain in this state unless the startup configuration includes the shutdown command. This behavior is consistent with all the other interfaces. When you create a BDI dynamically at command prompt, the default administrative state is down.

BDI Link State

A BDI maintains a link state that comprises of three states: administratively down, operationally down, and up. The link state of a BDI is derived from two independent inputs: the BDI administrative state set by the corresponding users and the fault indication state from the lower levels of the interface states. It defines a BDI link state based on the state of the two inputs.

Fault Indication State	BDI Admin	
{start emdash} {end emdash}	**Shutdown**	**No Shutdown**
No faults asserted	Admin-down	Up
At least one fault asserted	Admin-down	Operationally-Down

Bridge Domain Interface Statistics

For virtual interfaces, such as the bridge domain interface, protocol counters are periodically queried from the QFP.

When packets flow from a Layer 2 bridge domain network to a Layer 3 routing network through the bridge domain interface, the packets are treated as bridge domain interface input packets and bytes. When packets arrive at a Layer 3 interface and are forwarded through the bridge domain interface to a Layer 2 bridge domain, the packets are treated as output packets and bytes, and the counters are updated accordingly.

A BDI maintains a standard set of Layer 3 packet counters as the case with all Cisco IOS interfaces. Use the show interface command to view the Layer 3 packet counters.

The convention of the counters is relative to the Layer 3 cloud. For example, input refers to the traffic entry to the Layer 3 cloud from the Layer 2 BD, while output refers to the traffic exit from the Layer 3 cloud to the Layer 2 BD.

Use the **show interfaces accounting** command to display the statistics for the BDI status. Use the **show interface** <*if-name*> command to display the overall count of the packets and bytes that are transmitted and received.

Creating or Deleting a Bridge Domain Interface

When you define an interface or subinterface for a Cisco IOS router, you name it and specify how it is assigned an IP address. You can create a bridge domain interface before adding a bridge domain to the system. This new bridge domain interface will be activated after the associated bridge domain is configured.

Note When a bridge domain interface is created, a bridge domain is automatically created.

When you create the bridge domain interface and the bridge domain, the system maintains the required associations for mapping the bridge domain-bridge domain interface pair.

The mapping of bridge domain and bridge domain interface is maintained in the system. The bridge domain interface uses the index of the associated bridge domain to show the association.

Bridge Domain Interface Scalability

The following table lists the bridge domain interface scalability numbers, based on the type of Cisco 4000 Series ISR devices' Forwarding Processors (FPs).

Table 40: Bridge Domain Interface Scalability Numbers Based on the Type of Cisco 4000 Series ISR devices' Forwarding Processor

Description			0
Maximum bridge domain interfaces per router			

Bridge-Domain Virtual IP Interface

The Virtual IP Interface (VIF) feature helps to associate multiple BDI interfaces with a BD instance. The BD-VIF interface inherits all the existing L3 features of IOS logical IP interface.

Note You must configure every BD-VIF interface with a unique MAC address and it should belong to a different VRF.

The Virtual IP Interface (VIF) feature has the following limitations:

- BD-VIF interface does not support IP multicast.

- Number of BD-VIF interfaces with automatically generated MAC address varies on the basis of platforms.
- BD-VIF Interface does not support MPLS.
- The maximum number of BD-VIF interfaces per bridge-domain and the total number of BD-VIF interface for per system vary based on the type of platforms.

The maximum number of BD-VIF supported on different platforms varies:

- ASR 1000 supports maximum 100 BD-VIF for a Bridge Domain
- CSR 1000v supports maximum 16 BD-VIF for a Bridge Domain
- ISR 4000 support maximum 16 BD-VIF for a Bridge Domain

From Cisco IOS XE 17.7.1a release, BD-VIF supports Flexible Netflow (FNF).

How to Configure a Bridge Domain Interface

To configure a bridge domain interface, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface BDI** *{interface number}*
4. **encapsulation** *encapsulation dot1q <first-tag> [second-dot1q <second-tag>]*
5. Do one of the following:
6. **match security-group destination tag** *sgt-number*
7. **mac address** *{mac-address}*
8. **no shut**
9. **shut**

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	**interface BDI** *{interface number}* **Example:** `Router(config-if)# interface BDI3`	Specifies a bridge domain interface.

	Command or Action	Purpose
Step 4	**encapsulation** *encapsulation dot1q <first-tag> [second-dot1q <second-tag>]* **Example:** `Router(config-if)# encapsulation dot1Q 1 second-dot1q 2`	Defines the encapsulation type. The example shows how to define dot1q as the encapsulation type.

Step 5	Do one of the following:
	Example:
	ip address *ip-address mask*
	Example:
	Example:
	ipv6 address *{X:X:X:X::X* **link-local**| *X:X:X:X::X/prefix [***anycast** | **eui-64***]* | **autoconfig** *[***default***]}*
	Example:
	`Router(config-if)# ip address 10.2.2.1 255.255.255.0`
	Example:
	Example:
	`Router(config-if)# ipv6 address AB01:CD1:123:C::/64 eui-64`
		Specifies either the IPv4 or IPv6 address for the bridge domain interface.

Step 6	**match security-group destination tag** *sgt-number* **Example:** `Router(config-route-map)# match security-group destination tag 150`	Configures the value for security-group destination security tag.
Step 7	**mac address** *{mac-address}* **Example:** `Router(config-if)# mac-address 1.1.3`	Specifies the MAC address for the bridge domain interface.
Step 8	**no shut** **Example:** `Router(config-if)# no shut`	Enables the bridge domain interface.
Step 9	**shut** **Example:**	Disables the bridge domain interface.

	Command or Action	Purpose
	`Router(config-if)# shut`	

Example

The following example shows the configuration of a bridge domain interface at IP address 10.2.2.1 255.255.255.0:

```
Router# configure terminal
Router(config)# interface BDI3
Router(config-if)# encapsulation dot1Q 1 second-dot1q 2
Router(config-if)# ip address 10.2.2.1 255.255.255.0
Router(config-if)# mac-address 1.1.3
Router(config-if)# no shut
Router(config-if)# exit
```

Displaying and Verifying Bridge Domain Interface Configuration

SUMMARY STEPS

1. **enable**
2. **show interfaces bdi**
3. **show platform software interface fp active name**
4. **show platform hardware qfp active interface if-name**
5. **debug platform hardware qfp feature**
6. **platform trace runtime process forwarding-manager module**
7. **platform trace boottime process forwarding-manager module interfaces**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**show interfaces bdi** **Example:** `Router# show interfaces BDI3`	Displays the configuration summary of the corresponding BDI.
Step 3	**show platform software interface fp active name** **Example:** `Router# show platform software interface fp active name BDI4`	Displays the bridge domain interface configuration in a Forwarding Processor.

	Command or Action	Purpose
Step 4	**show platform hardware qfp active interface if-name** **Example:** `Router# show platform hardware qfp active interface if-name BDI4`	Displays the bridge domain interface configuration in a data path.
Step 5	**debug platform hardware qfp feature** **Example:** `Router# debug platform hardware qfp active feature l2bd client all`	The selected CPP L2BD Client debugging is on.
Step 6	**platform trace runtime process forwarding-manager module** **Example:** `Router(config)# platform trace runtime slot F0 bay 0 process forwarding-manager module interfaces level info`	Enables the Forwarding Manager Route Processor and Embedded Service Processor trace messages for the Forwarding Manager process.
Step 7	**platform trace boottime process forwarding-manager module interfaces** **Example:** `Router(config)# platform trace boottime slot R0 bay 1 process forwarding-manager forwarding-manager level max`	Enables the Forwarding Manager Route Processor and Embedded Service Processor trace messages for the Route Processor Forwarding Manager process during bootup.

What to do next

For additional information on the commands and the options available with each command, see the Cisco IOS Configuration Fundamentals Command Reference Guide.

Configuring Bridge-Domain Virtual IP Interface

```
enable
configure terminal
[no] interface BD-VIF interface-number
  [ [no] vrf forwarding vrf-name]
  [ [no] mac address mac-address]
  [ [no] ip address ip-address mask]
  [ [no] ipv6 address {X:X:X:X::X link-local| X:X:X:X::X/prefix [anycast | eui-64] |
autoconfig [default]}]

exit
```

To delete BD-VIF interface, use the 'no' form of the command.

Associating VIF Interface with a Bridge Domain

```
enable
configure terminal
bridge-domain bridge-domain number
[no] member BD-VIF interface-number
exit
```

To dissociate the VIF interface, use the 'no' form of the command.

Verifying Bridge-Domain Virtual IP Interface

All existing show commands for interface and IP interface can be used for the BD-VIF interface.

show interface bd-vif *bd-vif-id*

show ip interface bd-vif *bd-vif-id*

show bd-vif interfaces in fman-fp

show pla sof inter fp ac brief | i BD_VIF

Example Configuration Bridge-Domain Virtual IP Interface

```
Detail sample:

interface Port-channel1
mtu 9000
no ip address
 !Ethernet service endpoint one per neutron network
service instance 1756 ethernet
  description 4e8e5957-649f-477b-9e5b-f1f75b21c03c
  encapsulation dot1q 1756
  rewrite ingress tag pop 1 symmetric
  bridge-domain 1756
!
interface BD-VIF5001
no shutdown
vrf forwarding vrf5001
ip address 10.0.0.1 255.255.255.0
interface BD-VIF5002
no shutdown
vrf forwarding vrf5002
ip address 10.0.0.2 255.255.255.0

bridge-domain 1756
member Port-channel1 service-instance 1756
member bd-vif5001
member bd-vif5002
```

Configuring Flexible NetFlow over a Bridge Domain Virtual IP Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **interface** *type number*
4. {**ip** | **ipv6**}**flow monitor** *monitor-name* [**sampler** *sampler-name*] {**input** | **output**}
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device (config)# interface BD-VIF 100`	Specifies an interface and enters interface configuration mode. Enter the BD-VIF number.
Step 4	{**ip** | **ipv6**}**flow monitor** *monitor-name* [**sampler** *sampler-name*] {**input** | **output**} **Example:** `Device(config-if)# ip flow monitor FLOW-MONITOR-1 input`	Enables a Flexible NetFlow flow monitor for IP traffic that the router is receiving or transmitting on the interface.
Step 5	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and returns to privileged EXEC mode.

Examples: Flexible NetFlow over a Bridge Domain Virtual IP Interface

The following is a sample output for the **show platform hardware qfp active interface if-name** command showing the QFP information and flow direction for flow monitors. The table below provides the key to the CLI output.

Configuration	Output
ip flow monitor <monitor-name> input	IPV4_INPUT_FNF_FIRST IPV4_INPUT_FNF_FINAL
ip flow monitor <monitor-name> output	IPV4_BDI_OUTPUT_FNF_FINAL
ipv6 flow monitor <monitor-name> input	IPV6_INPUT_FNF_FIRST IPV6_INPUT_FNF_FINAL
ipv6 flow monitor <monitor-name> output	IPV6_BDI_OUTPUT_FNF_FINAL

```
Device# show run interface bd-vif2
Building configuration...

Current configuration: 227 bytes
!
interface BD-VIF2
vrf forwarding vrf1
ip flow monitor test1 input
ip flow monitor test1 output
ip address 10.11.11.11 255.255.255.0
ipv6 flow monitor test2 input
ipv6 flow monitor test2 output
ipv6 address 2001:DB8::1/32
end

Device# show platform hardware qfp active interface if-name BD-VIF 2
General interface information
  Interface Name: BD-VIF2
  Interface state: VALID
  Platform interface handle: 20
  QFP interface handle: 17
  Rx uidb: 262138
  Tx uidb: 262127
  Channel: 0
Interface Relationships

BGPPA/QPPB interface configuration information
  Ingress: BGPPA/QPPB not configured. flags: 0000
  Egress: BGPPA not configured. flags: 0000

ipv4_input enabled.
ipv4_output enabled.
ipv6_input enabled.
ipv6_output enabled.
layer2_input enabled.
layer2_output enabled.
ess_ac_input enabled.

Features Bound to Interface:
2 GIC FIA state
66 PUNT INJECT DB
70 cpp_l2bd_svr
43 icmp_svr
45 ipfrag_svr
46 ipreass_svr
47 ipv6reass_svr
44 icmp6_svr
58 stile
Protocol 0 - ipv4_input
FIA handle - CP:0x55a7f59df038 DP:0x3fff1000
  IPV4_INPUT_DST_LOOKUP_ISSUE (M)
  IPV4_INPUT_ARL_SANITY (M)
  IPV4_INPUT_SRC_LOOKUP_ISSUE
  IPV4_INPUT_DST_LOOKUP_CONSUME (M)
  IPV4_INPUT_SRC_LOOKUP_CONSUME
  IPV4_INPUT_FOR_US_MARTIAN (M)
  IPV4_INPUT_STILE_LEGACY
  IPV4_INPUT_FNF_FIRST
  IPV4_INPUT_LOOKUP_PROCESS (M)
  IPV4_INPUT_FNF_FINAL
  IPV4_INPUT_IPOPTIONS_PROCESS (M)
  IPV4_INPUT_GOTO_OUTPUT_FEATURE (M)
Protocol 1 - ipv4_output
FIA handle - CP:0x55a7f59df0d8 DP:0x3ffeff00
```

```
  IPV4_VFR_REFRAG (M)
  IPV4_OUTPUT_SRC_LOOKUP_ISSUE
  IPV4_OUTPUT_L2_REWRITE (M)
  IPV4_OUTPUT_SRC_LOOKUP_CONSUME
  IPV4_OUTPUT_STILE_LEGACY
  IPV4_OUTPUT_FRAG (M)
  IPV4_BDI_OUTPUT_FNF_FINAL.
  BDI_VLAN_TAG_ATTACH_AND_LAYER2_LOOKUP_GOTO
  LAYER2_BRIDGE
  BDI_OUTPUT_GOTO_OUTPUT_FEATURE
  IPV4_OUTPUT_DROP_POLICY (M)
  DEF_IF_DROP_FIA (M)
Protocol 6 - ipv6_input
FIA handle - CP:0x55a7f59dee58 DP:0x3fff4300
  IPV6_INPUT_SANITY_CHECK (M)
  IPV6_INPUT_DST_LOOKUP_ISSUE (M)
  IPV6_INPUT_SRC_LOOKUP_ISSUE
  IPV6_INPUT_ARL (M)
  IPV6_INPUT_DST_LOOKUP_CONT (M)
  IPV6_INPUT_SRC_LOOKUP_CONT
  IPV6_INPUT_DST_LOOKUP_CONSUME (M)
  IPV6_INPUT_SRC_LOOKUP_CONSUME
  IPV6_INPUT_STILE_LEGACY
  IPV6_INPUT_FNF_FIRST
  IPV6_INPUT_FOR_US (M)
  IPV6_INPUT_LOOKUP_PROCESS (M)
  IPV6_INPUT_FNF_FINAL
  IPV6_INPUT_LINK_LOCAL_CHECK (M)
  IPV6_INPUT_GOTO_OUTPUT_FEATURE (M)
Protocol 7 - ipv6_output
FIA handle - CP:0x55a7f59dee08 DP:0x3fff4b80
  IPV6_VFR_REFRAG (M)
  IPV6_OUTPUT_SRC_LOOKUP_ISSUE
  IPV6_OUTPUT_SRC_LOOKUP_CONT
  IPV6_OUTPUT_SRC_LOOKUP_CONSUME
  IPV6_OUTPUT_L2_REWRITE (M)
  IPV6_OUTPUT_STILE_LEGACY
  IPV6_OUTPUT_FRAG (M)
  IPV6_BDI_OUTPUT_FNF_FINAL
  BDI_VLAN_TAG_ATTACH_AND_LAYER2_LOOKUP_GOTO
  LAYER2_BRIDGE
  BDI_OUTPUT_GOTO_OUTPUT_FEATURE
  IPV6_OUTPUT_DROP_POLICY (M)
  DEF_IF_DROP_FIA (M)
□
```

The following is a sample out of the **show flow monitor** [[**name**] [**cache** [**format** {**csv** | **record** | **table**}]] [**statistics**]] command showing the cache output in record format.

```
Device# show flow monitor name FLOW-MONITOR-1 cache format record

Cache type:  Normal
Cache size:  1000
Current entries:  4
High Watermark:  4
Flows added:  101
Flows aged:  97
- Active timeout (1800 secs) 3
- Inactive timeout  (15 secs) 94
- Event aged  0
- Watermark aged  0
- Emergency aged
IPV4 DESTINATION ADDRESS:
198.51.100.1 0
ipv4 source address: 10.10.11.1
```

```
trns source port: 25
trns destination port: 25
counter bytes: 72840
counter packets: 1821
IPV4 DESTINATION ADDRESS: 198.51.100.2
ipv4 source address: 10.10.10.2
trns source port: 20
trns destination port: 20
counter bytes: 3913860
counter packets: 7326
IPV4 DESTINATION ADDRESS: 198.51.100.200
ipv4 source address: 192.168.67.6
trns source port: 0
trns destination port: 3073
counter bytes: 51072
counter packets: 1824


Device# show flow monitor name FLOW-MONITOR-2 cache format record

Cache type:  Normal
Cache size: 1000
Current entries: 2
High Watermark: 3
Flows added: 95
Flows aged: 93
- Active timeout (1800 secs) 0
- Inactive timeout (15 secs) 93
- Event aged 0
- Watermark aged 0
- Emergency aged 0
IPV6 DESTINATION ADDRESS: 2001:DB8:0:ABCD::1
ipv6 source address: 2001:DB8:0:ABCD::2
trns source port: 33572
trns destination port: 23
counter bytes: 19140
counter packets: 349
IPV6 DESTINATION ADDRESS: FF02::9
ipv6 source address: 2001:DB8::A8AA:BBFF:FEBB

trns source port: 521
trns destination port: 521
counter bytes: 92
counter packets: 1
```

The following is a sample out of the **show flow interface** command showing the flow status for an interface.

```
Device# show flow interface BD-VIF2001

Interface GigabitEthernet0/0/0
FNF: monitor: FLOW-MONITOR-1
direction: Input
traffic(ip): on
FNF: monitor: FLOW-MONITOR-2
direction:    Input traffic(ipv6): on

Device# show flow interface BD-VIF2002

Interface GigabitEthernet1/0/0
FNF: monitor: FLOW-MONITOR-1
direction: Output
traffic(ip): on
FNF: monitor: FLOW-MONITOR-2
direction:    Input traffic(ipv6): on
```

The following is a sample output of the **show platform hardware qfp active interface if-name | in FNF** command showing the QFP information and flow direction for flow monitors in Flexible NetFlow configuration. The table below provides the key to the CLI output.

Configuration	Output
ip flow monitor <monitor-name> input	IPV4_INPUT_FNF_FIRST IPV4_INPUT_FNF_FINAL
ip flow monitor <monitor-name> output	IPV4_BDI_OUTPUT_FNF_FINAL
ipv6 flow monitor <monitor-name> input	IPV6_INPUT_FNF_FIRST IPV6_INPUT_FNF_FINAL
ipv6 flow monitor <monitor-name> output	IPV6_BDI_OUTPUT_FNF_FINAL

```
Device# show run interface bd-vif2
Building configuration...

Current configuration : 227 bytes
!
interface BD-VIF2
vrf forwarding vrf1
ip flow monitor test1 input
ip flow monitor test1 output
ip address 10.11.11.11 255.255.255.0
ipv6 flow monitor test2 input
ipv6 flow monitor test2 output
ipv6 address 2001::8/64
end
 Device# show platform hardware qfp active interface if-name BD-VIF 2  | in FNF
  IPV4_INPUT_FNF_FIRST
  IPV4_INPUT_FNF_FINAL
  IPV4_BDI_OUTPUT_FNF_FINAL.
  IPV6_INPUT_FNF_FIRST
  IPV6_INPUT_FNF_FINAL
  IPV6_BDI_OUTPUT_FNF_FINAL
```

The **clear flow monitor name** *monitor-name* [**cache** [**force-export**] | **force-export** | **statistics**] command clears a Flexible NetFlow flow monitor, flow monitor cache, or flow monitor statistics, and can be used to force the export of the data in the flow monitor cache.

For more details on configuring Flexible NetFlow, see the Flexible NetFlow Configuration Guide, Cisco IOS XE 17.

Additional References

Related Documents

Related Topic	Document Title
Configuring Ethernet Virtual Connections on the Cisco ASR 1000 Series Aggregation Services Routers	Carrier Ethernet Configuration Guide
EVC Quality of Service	http://www.cisco.com/en/US/docs/ios/ios_xe/qos/configuration/guide/qos_evc_xe.html

MIBs

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

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.	https://www.cisco.com/c/en_in/support/index.html

Feature Information for Configuring Bridge Domain Interfaces

The following table lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note The table below 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.

Table 41: Feature Information for Configuring Bridge Domain Interfaces

Feature Name	Releases	Feature Information
Configuring Bridge Domain Interface	Cisco IOS XE Cupertino 17.7.1a	This feature was introduced on the Cisco 4000 Series ISR devices.
Bridge-Domain Virtual IP Interface	Cisco IOS XE Cupertino 17.7.1a	This feature was introduced on the Cisco 4000 Series ISR devices. The Bridge-Domain Virtual IP Interface (VIF) now connects multiple Bridge Domain Interfaces (BDI) with a single BD instance so that each IP subnet within an L2 network can be associated with a single VRF.
Flexible NetFlow (FNF) on Bridge-Domain Virtual IP Interface (BD-VIF)	Cisco IOS XE Cupertino 17.7.1a	This feature was introduced on the Cisco 4000 Series ISR devices. The following command was introduced: **{ip | ipv6} flow monitor** *monitor-name* **[sampler** *sampler-name*] **{input | output}**

CHAPTER 23

Managing Cisco Enhanced Services and Network Interface Modules

The router supports Cisco Enhanced Services Modules (SMs) and Cisco Network Interface Modules (NIMs). The modules are inserted into the router using an adapter, or carrier card, into various slots. For more information, see the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

The following sections are included in this chapter:

Information About Cisco Enhanced Services and Network Interface Modules

The router configures, manages, and controls the supported Cisco Enhanced Services Modules (SMs) and Network Interface Modules (NIMs) using the module management facility built in its architecture. This new centralized module management facility provides a common way to control and monitor all the modules in the system regardless of their type and application. All Cisco Enhanced Service and Network Interface Modules supported on your router use standard IP protocols to interact with the host router. Cisco IOS software uses alien data path integration to switch between the modules.

- Modules Supported, on page 378
- Network Interface Modules, on page 378
- Enhanced Service Modules, on page 380

Modules Supported

For information about the interfaces and modules supported by the Cisco ISR 4400 series and Cisco ISR 4300 series routers, see http://www.cisco.com/c/en/us/products/routers/4000-series-integrated-services-routers-isr/relevant-interfaces-and-modules.html.

Network Interface Modules

The following Network Interface Modules are supported:

Cisco Fourth-Generation LTE Network Interface Module

Cisco 4G LTE NIM addresses the modular 4G LTE cellular connectivity on the Cisco 4000 Series ISRs. This is the first wireless NIM, though it is not the first wireless module in the ISR product line. The closest modular card to Cisco 4G LTE NIM is the Cisco EHWIC 4G LTE, which accepts a single LTE modem. Cisco 4G LTE NIM is feature-compatible with Cisco EHWIC 4G LTE. For more information, see the Cisco Fourth-Generation LTE Network Interface Module Software Configuration Guide.

Cisco 4-Port and 8-Port Layer 2 Gigabit EtherSwitch Network Interface Module

The Cisco 4-Port and 8-Port Layer 2 Gigabit EtherSwitch Network Interface Module (NIM) integrates the Layer 2 features and provides a 1-Gbps connection to the multigigabit fabric (MGF) for intermodule communication. For more information on configuring the Cisco 4-Port and 8-Port Layer 2 Gigabit EtherSwitch NIM, see http://www.cisco.com/c/en/us/td/docs/routers/access/interfaces/NIM/software/configuration/guide/4_8PortGENIM.html.

Cisco Fourth-Generation T1/E1 Voice and WAN Network Interface Module

The Cisco Fourth-Generation T1/E1 Voice and WAN Network Interface Module (NIM) is inserted into the NIM slot of the router and provides data and voice support on T1/E1 trunks. To support voice-related and other DSP features, the Cisco PVDM4 (Cisco Packet Voice Digital Signal Processor Module) is also required. See the following documents for more information:

- Installing the Cisco Fourth-Generation T1/E1 Voice and WAN Network Interface Module
- Configuring the Cisco Fourth-Generation T1/E1 Voice and WAN Network Interface Module

- Installing the Cisco PVDM4

Cisco SSD/HDD Carrier Card NIM

The router supports a single Cisco SSD and HDD Carrier Card NIM, which must be placed in slot 0 and subslot 1, 2, or 3.

A Cisco SSD/HDD Carrier Card NIM can be one of the following:

- Cisco SSD Carrier Card NIM—Supports one or two Solid-State Drives (SSDs).
- Cisco HDD Carrier Card NIM—Supports one Hard Disk Drive (HDD).

Note When ISR-WAAS is operational, do not perform online insertion or replacement (OIR) of NIM-SSD and NIM-HDD.

For more information on the hardware characteristics of the SSD/HDD Carrier Card NIM, see the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

For more information on deactivating or reactivating a SSD/HDD Carrier Card NIM, see Deactivating and Reactivating an SSD/HDD Carrier Card NIM, on page 385.

Cisco 1-, 2-, and 4-Port Serial NIM

The Cisco 1-, 2-, and 4-port Serial NIMs are multi-protocol synchronous serial network interface modules (NIMs) supported on the Cisco 4400 Series ISRs. The Cisco 1-, 2-, and 4-port Serial NIMs expand the capabilities of the router to provide connectivity for synchronous interfaces in a wide range of applications including up to 8Mbps data rate for high speed high-level data link control (HDLC). These capabilities can be utilized as Point-to-Point Cisco HDLC WAN interface or frame relay interface. The Cisco 1-, 2-, and 4-port Serial NIMs have their own serial communication controllers (SCC) and they do not rely on the host router for SCCs. For further information on configuring this NIM, see the Configuring the Cisco 1-, 2-, and 4-port Serial Network Interface Modules for the Cisco 4400 Series ISRs document.

Upgrading the SSD or HDD Firmware

You can upgrade the firmware for the SSD or HDD using the **upgrade hw-programmable module filename bootflash:***filename slot/sub-slot* command.

A typical *filename* has the form: *nim_ssd_manufacturer_firmware-version-number.bin*

The firmware file can also be available in other locations other than **bootflash:**

For example, you can provide any one of the following locations in place of **bootflash:***filename*:

- **flash:***filename*
- **harddisk:***filename*
- **usb1:***filename*

Note For a Cisco SSD carrier card NIM or Cisco HDD carrier card NIM, only slot 0 and one of the subslots 1, 2, or 3 must be used.

The following example shows how to upgrade a Micron P400m disk to firmware revision 200 using the **upgrade hw-programmable module filename bootflash:***filename slot/sub-slot* command:

```
Router# upgrade hw-programmable module filename bootflash:nim_ssd_Micr nP400m_E200.bin
Info: Trying to upgrade Module in 0/3 with nim_ssd_MicronP400m_E200.bin
Info: Current NIM-SSD disk config.
Info: Disk1: rev: 0200 model: MicronP400m-MTFDDAK200MAN
Info: Disk2: rev: 0200 model: MicronP400m-MTFDDAK200MAN
/dev/sde:
fwdownload: xfer_mode=3 min=1 max=255 size=512
..........................................................................................
 Done.
/dev/sdf:
fwdownload: xfer_mode=3 min=1 max=255 size=512
..........................................................................................
 Done.
Info: Performing post upgrade check ......
Info: Upgrade to Firmware version E200 on disk1 successful.
Info: Upgrade to Firmware version E200 on disk2 successful.
Info: Current NIM-SSD disk config.
Info: Disk1: rev: E200 model: MicronP400m
```

Error Monitoring

The drives in the Cisco SDD/HDD Carrier Card NIM are monitored for SMART errors. If a SMART error occurs, a Cisco IOS error message is displayed, as shown in the following example:

```
%IOSXE-5-PLATFORM:logger: INFO:/dev/sde:SMART error present:please do
'more bootflash:/tracelogs/smart_errors.log'.
```

You can find additional information in the error log at: bootflash:/tracelogs/smart_errors.log

Enhanced Service Modules

The following service modules are supported on the router:

- Cisco SM-1 T3/E3 Service Module, on page 380
- Cisco UCS E-Series Server, on page 381
- Cisco SM-X Layer 2/3 EtherSwitch Service Module, on page 381
- Cisco 6-Port GE SFP Service Module, on page 381

Cisco SM-1 T3/E3 Service Module

For more information, see the Cisco SM-1T3/E3 Enhanced Service Module Configuration Guide.

Cisco UCS E-Series Server

For more information, see the documentation listed in the Cisco UCS E-Series Server Roadmap.

Cisco SM-X Layer 2/3 EtherSwitch Service Module

This module provides the following features:

- Integration of Layer 2 and Layer 3 switching features and the ability of the router to use the Cisco SM-X Layer 2/3 ESM (16-port and 24-port) as an independent Layer 3 switch.
- 1 Gbps connection to the multigigabit fabric (MGF) for intermodule communication without burdening the CPU of the router.
- Up to 30 watts of power per port with the robust Power over Ethernet Plus (PoE+) feature along with IEEE 802.3AE Media Access Control Security (MACSec) port-based, hop-to-hop, encryption, and Cisco TrustSec.

For more information, see the following documents:

- Cisco SM-X Layer 2/3 EtherSwitch Service Module Configuration Guide for Cisco 4451-X ISR
- Connecting Cisco SM-X Layer 2/3 EtherSwitch Service Module to the Network

Cisco 6-Port GE SFP Service Module

The Cisco 6-port GE SFP service module is a Gigabit Ethernet module that can be inserted into the router's SM slot to provide Gigabit Ethernet features on routable external interfaces. For more information about configuring this service module, see the Software Configuration Guide for the Cisco 6-port GE SFP Service Module.

Cisco 4-port GE SFP and 1-port 10 GE SFP Service Module

The Cisco 4-port GE SFP and 1-port 10 GE SFP Service Module (SM-X-4x1GE-1x10GE) is software-configurable high-speed connectivity routing port service module for the Cisco ISR 4400 Series routers. This service module provides increased density of Ethernet interfaces on the Cisco ISR 4400 Series routers. For further information on configuring this service module, see: the Software Configuration Guide for the Cisco 6-port GE SFP Service Module and Cisco 4-port GE SFP and 1-port 10 GE SFP Service Module

Cisco 1GE-CU-SFP and 2GE-CU-SFP Network Interface Modules

The Cisco 1GE-CU-SFP and 2GE-CU-SFP Network Interface Modules (NIMs) are software-configurable high-speed connectivity routing port network interface modules for the Cisco 4000 and Cisco ISR 4300 Series Integrated Services Routers (ISR). These network interface modules provide increased density of Ethernet interfaces on the Cisco 4000 ISR. For further information on configuring this NIM, see the Configuring the Cisco 1GE-CU-SFP and 2GE-CU-SFP Network Interface Modules in Cisco 4000 Series Integrated Services Routers.

Note Cisco 4221 ISR does not support 2GE-CU-SFP Network Interface Module.

Implementing SMs and NIMs on Your Router

Downloading the Module Firmware

Module firmware must be loaded to the router to be able to use a service module. For more information, see Installing a Firmware Subpackage, on page 124.

The modules connect to the RP via the internal eth0 interface to download the firmware. Initially, the module gets an IP address for itself via BOOTP. The BOOTP also provides the address of the TFTP server used to download the image. After the image is loaded and the module is booted, the module provides an IP address for the running image via DHCP.

Installing SMs and NIMs

For more information, see "Installing and Removing NIMs and SMs" in the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

Accessing Your Module Through a Console Connection or Telnet

Before you can access the modules, you must connect to the host router through the router console or through Telnet. After you are connected to the router, you must configure an IP address on the Gigabit Ethernet interface connected to your module. Open a session to your module using the **hw-module session** command in privileged EXEC mode on the router.

To establish a connection to the module, connect to the router console using Telnet or Secure Shell (SSH) and open a session to the switch using the **hw-module session** *slot/subslot* command in privileged EXEC mode on the router.

Use the following configuration examples to establish a connection:

- The following example shows how to open a session from the router using the **hw-module session** command:

```
Router# hw-module session slot/card
Router# hw-module session 0/1 endpoint 0

Establishing session connect to subslot 0/1
```

- The following example shows how to exit a session from the router, by pressing **Ctrl-A** followed by **Ctrl-Q** on your keyboard:

```
type ^a^q
picocom v1.4

port is        : /dev/ttyDASH2
flowcontrol    : none
baudrate is    : 9600
parity is      : none
databits are   : 8
escape is      : C-a
noinit is      : no
noreset is     : no
nolock is      : yes
send_cmd is    : ascii_xfr -s -v -l10
receive_cmd is : rz -vv
```

Online Insertion and Removal

The router supports online insertion and removal (OIR) of Cisco Enhanced Services Modules and Cisco Network Interface Modules. You can perform the following tasks using the OIR function:

Note When ISR-WAAS is operational, do not perform online insertion or replacement (OIR).

Preparing for Online Removal of a Module

The router supports the OIR of a module, independent of removing another module installed in your router. This means that an active module can remain installed in your router, while you remove another module from one of the subslots. If you are not planning to immediately replace a module, ensure that you install a blank filler plate in the subslot.

Deactivating a Module

A module can be removed from the router without first being deactivated. However, we recommend that you perform a graceful deactivation (or graceful power down) of the module before removing it. To perform a graceful deactivation, use the **hw-module subslot** *slot/subslot* **stop** command in EXEC mode.

Note When you are preparing for an OIR of a module, it is not necessary to independently shut down each of the interfaces before deactivating the module. The **hw-module subslot** *slot/subslot* **stop** command in EXEC mode automatically stops traffic on the interfaces and deactivates them along with the module in preparation for OIR. Similarly, you do not have to independently restart any of the interfaces on a module after OIR.

The following example shows how to use the **show facility-alarm status** command to verify if any critical alarm is generated when a module is removed from the system:

```
Router# show facility-alarm status
System Totals  Critical: 5  Major: 1  Minor: 0

Source                  Severity      Description [Index]
------                  --------      -------------------
Power Supply Bay 1      CRITICAL      Power Supply/FAN Module Missing [0]
GigabitEthernet0/0/0    CRITICAL      Physical Port Link Down [1]
GigabitEthernet0/0/1    CRITICAL      Physical Port Link Down [1]
GigabitEthernet0/0/2    CRITICAL      Physical Port Link Down [1]
GigabitEthernet0/0/3    CRITICAL      Physical Port Link Down [1]
xcvr container 0/0/0    INFO          Transceiver Missing [0]
xcvr container 0/0/1    INFO          Transceiver Missing [0]
xcvr container 0/0/2    INFO          Transceiver Missing [0]
xcvr container 0/0/3    INFO          Transceiver Missing [0]
V: 1.0v PCH R0/18       MAJOR         Volt Above Normal [3]
```

Note A critical alarm (Active Card Removed OIR Alarm) is generated even if a module is removed after performing graceful deactivation.

Deactivating Modules and Interfaces in Different Command Modes

You can deactivate a module and its interfaces using the **hw-module subslot** command in one of the following modes:

- If you choose to deactivate your module and its interfaces by executing the **hw-module subslot** *slot/subslot* **shutdown unpowered** command in global configuration mode, you are able to change the configuration in such a way that no matter how many times the router is rebooted, the module does not boot. This command is useful when you need to shut down a module located in a remote location and ensure that it does not boot automatically when the router is rebooted.
- If you choose to use the **hw-module subslot** *slot/subslot* **stop** command in EXEC mode, you cause the module to gracefully shut down. The module is rebooted when the **hw-module subslot** *slot/subslot* **start** command is executed.

To deactivate a module and all of its interfaces before removing the module, use one of the following commands in global configuration mode.

Procedure

	Command or Action	Purpose
Step 1	**hw-module subslot** *slot/subslot* **shutdown unpowered** **Example:**	Deactivates the module located in the specified slot and subslot of the router, where:

	Command or Action	Purpose
	Router# **hw-module subslot 0/2 shutdown unpowered**	• *slot*—Specifies the chassis slot number where the module is installed. • *subslot*—Specifies the subslot number of the chassis where the module is installed. • **shutdown**—Shuts down the specified module. • **unpowered**—Removes all interfaces on the module from the running configuration and the module is powered off.
Step 2	**hw-module subslot** *slot/subslot* [**reload** | **stop** | **start**] **Example:** Router# **hw-module subslot 0/2 stop**	Deactivates the module in the specified slot and subslot, where: • *slot*—Specifies the chassis slot number where the module is installed. • *subslot*—Specifies the subslot number of the chassis where the module is installed. • **reload**—Stops and restarts the specified module. • **stop**—Removes all interfaces from the module and the module is powered off. • **start**—Powers on the module similar to a physically inserted module in the specified slot. The module firmware reboots and the entire module initialization sequence is executed in the IOSd and Input/Output Module daemon (IOMd) processes.

Deactivating and Reactivating an SSD/HDD Carrier Card NIM

The following restrictions apply:

- Deactivating or reactivating an SSD/HDD Carrier Card NIM without an SSD or HDD disk is not supported.
- Only a single (SSD or HDD) Carrier Card NIM can be plugged into a bay. If you plug an additional (SSD or HDD) Carrier Card NIM into another bay, the module powers down and kernel, log, or error messages are displayed on the Cisco IOS console. In rare cases, the file system may get corrupted on the additional drive.

Caution Deactivation of an SSD/HDD Carrier Card NIM may cause loss of data.

To deactivate an SSD/HDD Carrier Card NIM, perform the following steps:

Procedure

	Command or Action	Purpose
Step 1	**virtual-service** *name* **Example:** `Router(config)# virtual-service my-kwaas-instance`	Identifies the kWAAS service (by name), supported on your router, in preparation for the router to be shut down by the **no activate** command. We recommend that you use this command before reseating or replacing an SSD or HDD.
Step 2	**no activate** **Example:** `Router(config-virt-serv)# no activate`	Shuts down the kWAAS instance on your router. kWAAS services remain installed. The service will have to be reactivated after the HDD/SSD NIM (module) is restarted.

Step 3	**hw-module subslot** *slot/subslot* [**reload** | **stop** | **start**]
	Example:
	`Router# hw-module subslot 0/2 stop`
	`Proceed with stop of module? [confirm]`
	`Router#`
	`*Mar 6 15:13:23.997: %SPA_OIR-6-OFFLINECARD: SPA (NIM-SSD) offline in subslot 0/2`
	`...`
		Deactivates or reactivates the module in the specified slot and subslot.
		• *slot*—The chassis slot number where the module is installed.
		• *subslot*—The subslot number of the chassis where the module is installed.
		• **reload**—Deactivates and reactivates (stops and restarts) the specified module.
		• **stop**—Removes all interfaces from the module and the module is powered off.
		• **start**—Powers on the module similar to a physically inserted module in the specified slot. The module firmware reboots and the entire module initialization sequence is executed in the IOSd and IOMd processes.

Step 4	Wait for the EN (Enable) LED to turn off, and then remove the SSD/HDD Carrier Card NIM.	

Reactivating a Module

If, after deactivating a module using the **hw-module subslot** *slot/subslot* **stop** command, you want to reactivate it without performing an OIR, use one of the following commands (in privileged EXEC mode):

- **hw-module subslot** *slot/subslot* **start**
- **hw-module subslot** *slot/subslot* **reload**

Verifying the Deactivation and Activation of a Module

When you deactivate a module, the corresponding interfaces are also deactivated. This means that these interfaces will no longer appear in the output of the **show interface** command.

1. To verify the deactivation of a module, enter the **show hw-module subslot all oir** command in privileged EXEC configuration mode.

   Observe the "Operational Status" field associated with the module that you want to verify. In the following example, the module located in subslot 1 of the router is administratively down.

```
Router# show hw-module subslot all oir

Module         Model              Operational Status

-------------- ------------------ -------------------------

subslot 0/0    ISR4451-4X1GE      ok
subslot 1/0    SM-X-T1/E1         ok
```

2. To verify activation and proper operation of a module, enter the **show hw-module subslot all oir** command and observe "ok" in the **Operational Status** field as shown in the following example:

```
Router# show hw-module subslot all oir

Module         Model               Operational Status

-------------- ------------------  -------------------------

subslot 0/1    NIM-8MFT-T1/E1      ok
subslot 1/0    SM-X T1/E1          ok

Router# show platform hardware backplaneswitch-manager R0 status
slot  bay   port   enable   link status   speed(Mbps)   duplex   autoneg   pause_tx
pause_rx   mtu
--------------------------------------------------------------------------------------------------------
0      0     CP     True      Up           1000          Full    ENABLED    ENABLED
ENABLED   10240
1      0     GE1    True      Up           1000          Full    DISABLED   ENABLED
ENABLED   10240
1      0     GE0    True      Up           1000          Full    DISABLED   ENABLED
ENABLED   10240
2      0     GE1    True      Up           1000          Full    DISABLED   ENABLED
ENABLED   10240
2      0     GE0    True      Up           1000          Full    DISABLED   ENABLED
ENABLED   10240
0      1     GE1    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      1     GE0    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      2     GE1    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      2     GE0    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      3     GE1    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      3     GE0    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      4     GE1    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      4     GE0    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      0     FFP    True       Up         10000          Full    ENABLED    DISABLED
DISABLED  10240
slot  bay   port            mac    vid   modid     flags - Layer 2
------------------------------------------------------------------------------
0      0    FFP  2c54.2dd2.661b   2351       1              0x20
0      0    FFP  2c54.2dd2.661b   2352       1              0x20
0      0     CP  2c54.2dd2.661e   2351       0              0xC60
0      0     CP  2c54.2dd2.661e   2352       0              0x20
1      0    GE0  58bf.ea3a.00f6   2350       0              0x460
0      0    FFP  2c54.2dd2.661b   2350       1              0x20
1      0    GE0  58bf.ea3a.00f6   2352       0              0x20
0      0     CP  2c54.2dd2.661e   2350       0              0x20
1      0    GE0  58bf.ea3a.00f6   2351       0              0xC60
```

```
Port block masks: rows=from port, columns=to port, u=unknown unicast, m=unknown multicast,
 b=broadcast, A=all

           CP    FFP  1/0/1  1/0/0  2/0/1  2/0/0  0/1/1  0/1/0  0/2/1  0/2/0  0/3/1
0/3/0  0/4/1 0/4/0 drops
-------------------------------------------------------------------------------------------
CP          -     A     um     um     um     um     um     um     um     um     um
um     um     um     1
FFP         A     -     -      -      -      -      -      -      -      -      -
 -      -      -     0
1/0/1      um    umb    -     umb    umb    umb    umb    umb    umb    umb    umb
umb    umb    umb     0
1/0/0      um    umb   umb     -     umb    umb    umb    umb    umb    umb    umb
umb    umb    umb     6
2/0/1      um    umb   umb    umb     -     umb    umb    umb    umb    umb    umb
umb    umb    umb     0
2/0/0      um    umb   umb    umb    umb     -     umb    umb    umb    umb    umb
umb    umb    umb     6
0/1/1      um    umb   umb    umb    umb    umb     -     umb    umb    umb    umb
umb    umb    umb     0
0/1/0      um    umb   umb    umb    umb    umb    umb     -     umb    umb    umb
umb    umb    umb     0
0/2/1      um    umb   umb    umb    umb    umb    umb    umb     -     umb    umb
umb    umb    umb     0
0/2/0      um    umb   umb    umb    umb    umb    umb    umb    umb     -     umb
umb    umb    umb     0
0/3/1      um    umb   umb    umb    umb    umb    umb    umb    umb    umb     -
umb    umb    umb     0
0/3/0      um    umb   umb    umb    umb    umb    umb    umb    umb    umb    umb
-     umb    umb     0
0/4/1      um    umb   umb    umb    umb    umb    umb    umb    umb    umb    umb
umb     -     umb     0
0/4/0      um    umb   umb    umb    umb    umb    umb    umb    umb    umb    umb
umb    umb     -      0

Port VLAN membership: [untagged vlan] U=untagged T=tagged <VLAN range begin>-<VLAN range
 end>

    CP [2352] U:0001-0001 T:0002-2351 U:2352-2352 T:2353-4095
   FFP [2352] T:0001-4095
 1/0/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 1/0/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 2/0/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 2/0/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/1/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/1/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/2/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/2/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/3/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/3/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/4/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/4/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
```

show platform hardware backplaneswitch-manager rp active ffp statistics: Example

```
Router# show platform hardware backplaneswitch-manager rp active ffp statistics
Broadcom 10G port(e.g: FFP) status:
                          Rx pkts         Rx Bytes          Tx Pkts         Tx Bytes
------------------------------------------------------------------------------------------
All                           0                0                0                0
  =64                         0                                 0
  65~127                      0                                 0
```

```
  128~255                     0                                  0
  256~511                     0                                  0
  512~1023                    0                                  0
  1024~1518                   0                                  0
  1519~2047                   0                                  0
  2048~4095                   0                                  0
  4096~9216                   0                                  0
  9217~16383                  0                                  0
  Max                         0                                  0
Good                          0                                  0
  CoS 0                                                          0                0
  CoS 1                                                          0                0
  CoS 2                                                          0                0
  CoS 3                                                          0                0
  CoS 4                                                          0                0
  CoS 5                                                          0                0
  CoS 6                                                          0                0
  CoS 7                                                          0                0
  Unicast                     0                                  0
  Multicast                   0                                  0
  Broadcast                   0                                  0
  Control                     0
Errored
  FCS                         0                                  0
  Undersize                   0
  Ether len                   0
  Fragment                    0                                  0
  Jabber                      0
  MTU ck, good                0
  MTU ck, bad                 0
  Tx underflow                                                                    0
  err symbol                  0
  frame err                   0
  junk                        0
Drops
  CoS 0                                                          0                0
  CoS 1                                                          0                0
  CoS 2                                                          0                0
  CoS 3                                                          0                0
  CoS 4                                                          0                0
  CoS 5                                                          0                0
  CoS 6                                                          0                0
  CoS 7                                                          0                0
  STP                         0
  backpress                   0
  congest                     0                0
  purge/cell                  0
  no destination              0
Pause PFC                     0                                  0
  CoS 0                       0
  CoS 1                       0
  CoS 2                       0
  CoS 3                       0
  CoS 4                       0
  CoS 5                       0
  CoS 6                       0
  CoS 7                       0
```

Managing Modules and Interfaces

The router supports various modules. For a list of supported modules, see Modules Supported, on page 378. The module management process involves bringing up the modules so that their resources can be utilized. This process consists of tasks such as module detection, authentication, configuration by clients, status reporting, and recovery. For detailed information about module configuration, see the module documentation referred to in the Documentation Roadmap for the Cisco 4000 Series Integrated Services Routers.

For a list of small-form-factor pluggable (SFP) modules supported on your router, see the "Installing and Upgrading Internal Modules and FRUs" section in the Hardware Installation Guide for the Cisco 4000 Series Integrated Services Routers.

The following sections provide additional information on managing the modules and interfaces:

- Managing Module Interfaces, on page 390
- Managing Modules and Interfaces Using Backplane Switch, on page 390

Managing Module Interfaces

After a module is in service, you can control and monitor its module interface. Interface management includes configuring clients with **shut** or **no shut** commands and reporting on the state of the interface and the interface-level statistics.

Monitor the module status and other statistical information using the **show** commands listed in Monitoring and Troubleshooting Modules and Interfaces, on page 393.

Managing Modules and Interfaces Using Backplane Switch

- Backplane Ethernet Switch, on page 390
- Viewing Module and Interface Card Status on a Router, on page 391
- Viewing Backplane Switch Statistics, on page 391
- Viewing Backplane Switch Port Statistics, on page 392
- Viewing Slot Assignments, on page 393

Backplane Ethernet Switch

The backplane Ethernet switch on your router provides connectivity to Enhanced Service Modules and Network Interface Modules (NIMs). The backplane Ethernet switch facilitates all packet transfers between the host router and its pluggable modules.

The backplane Ethernet switch act as a manager for the host router and controls the module and exchanges logical flow-control information with the module to ensure accurate feedback to the router features. See Managing Modules and Interfaces, on page 390 for more information. The backplane Ethernet switch also facilitates control plane traffic flow from the host router to the modules. The backplane switch manages modules and interface cards and is used to communicate with the modules. Module drivers integrate with the backplane switch to configure packet flow and control traffic buffering.

You are not required to perform any configuration tasks on the backplane switch; all the configurations are performed from the module, which may or may not lead to changes on the backplane switch. For more information on installing an adapter, see the Hardware Installation Guide for the Cisco ISR 4000 Series Integrated Services Routers.

Note Layer 2 protocols, such as the IEEE 802.1D Spanning Tree Protocol (STP), are not supported in the backplane Ethernet switch.

Viewing Module and Interface Card Status on a Router

You can view the module and interface card details using the **show platform** command in privileged EXEC mode.

The following example shows the sample output for the **show platform** command:

```
Router# show platform
Chassis type: ISR4451/K9

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ISR4451/K9          ok                    15:57:33
 0/0      ISR4451-4X1GE       ok                    15:55:24
 0/3      NIM-SSD             ok                    15:55:24
1         ISR4451/K9          ok                    15:57:33
 1/0      SM-1T3/E3           ok                    15:55:24
2         ISR4451/K9          ok                    15:57:33
 2/0      SM-1T3/E3           ok                    15:55:24
R0        ISR4451/K9          ok, active            15:57:33
F0        ISR4451-FP          ok, active            15:57:33
P0        Unknown             ps, fail              never
P1        XXX-XXXX-XX         ok                    15:56:58
P2        ACS-4450-ASSY       ok                    15:56:58

Slot      CPLD Version        Firmware Version
--------- ------------------- ---------------------------------------
0         12090323            15.3(01r)S            [ciscouser-ISRRO...
1         12090323            15.3(01r)S            [ciscouser-ISRRO...
2         12090323            15.3(01r)S            [ciscouser-ISRRO...
R0        12090323            15.3(01r)S            [ciscouser-ISRRO...
F0        12090323            15.3(01r)S            [ciscouser-ISRRO...
```

Viewing Backplane Switch Statistics

Statistics reports for each slot show incoming and outgoing packets or bytes. You can use the information to check traffic flow on the various ports of the backplane switch. The following example shows a sample output for the **show platform hardware backplaneswitch-manager rp active summary** command:

```
Router# show platform hardware backplaneswitch-manager rp active summary
slot     bay        port        InBytes          InPkts         OutBytes          OutPkts
-----------------------------------------------------------------------------------------
   0        0         CP           6242         9361008             6241           403209
   1        0        GE1              0               0                0                0
   1        0        GE0           6306          407477             6241          9360934
   2        0        GE1              0               0                0                0
   2        0        GE0              0               0                0                0
   0        1        GE1              0               0                0                0
   0        1        GE0              0               0                0                0
   0        2        GE1              0               0                0                0
```

```
  0       2       GE0             0               0               0               0
  0       3       GE1             0               0               0               0
  0       3       GE0             0               0               0               0
  0       4       GE1             0               0               0               0
  0       4       GE0             0               0               0               0
  0       0       FFP             0               0               0               0
  0       0       FFP             0               0               0               0
```

Viewing Backplane Switch Port Statistics

You can view statistical information related to the port connected to the backplane switch using the **show platform hardware backplaneswitch-manager rp active subslot GEO statistics** command. The following example displays statistical information related to the backplane switch and ports connected to it:

```
Router# show platform hardware backplaneswitch-manager rp active subslot 1/0 GE0 statistics
Broadcom 1G port(e.g: NIM, ESM, CP) status:
                          Rx pkts         Rx Bytes          Tx Pkts         Tx Bytes
-------------------------------------------------------------------------------------
All                          6306           407477             6241          9360934
  =64                        6237                                72
  65~127                       66                                 3
  128~255                       0                                 0
  256~511                       1                                 3
  512~1023                      2                                 0
  1024~1518                     0                              6163
  1519~2047                     0                                 0
  2048~4095                     0                                 0
  4096~9216                     0                                 0
Good                         6306                              6241
  CoS 0                                                        6171          9356426
  CoS 1                                                           0                0
  CoS 2                                                           0                0
  CoS 3                                                           0                0
  CoS 4                                                           0                0
  CoS 5                                                           0                0
  CoS 6                                                          70             4508
  CoS 7                                                           0                0
  Unicast                    6294                              6241
  Multicast                     6                                 0
  Broadcast                     6                                 0
  Control                       0                                 0
  VLAN                          0                                 0
Errored
  FCS                           0                                 0
  Runts                         0                0
  Undersize                     0
  Ether len                     0
  Fragment                      0                                 0
  Jabber                        0                                 0
  MTU                           0
Drops
  CoS 0                                                           0                0
  CoS 1                                                           0                0
  CoS 2                                                           0                0
  CoS 3                                                           0                0
  CoS 4                                                           0                0
  CoS 5                                                           0                0
  CoS 6                                                           0                0
  CoS 7                                                           0                0
  STP                           0
  backpress                     0
```

```
  congest                       0                0
  purge/cell                    0
  no destination               65
Pause                           0                                 0
```

Viewing Slot Assignments

Use the **show inventory** command in privileged EXEC mode to view the slot assignments, as shown in the following example:

```
Router# show inventory
NAME: "Chassis", DESCR: "Cisco ISR4451 Chassis"
PID: ISR4451/K9        , VID: V01, SN: FGL163910CM

NAME: "Power Supply Module 1", DESCR: "Cisco 4451-X ISR 450W AC Power Supply"
PID: XXX-XXXX-XX       , VID: XXX, SN: DCA1623X05N

NAME: "Fan Tray", DESCR: "Cisco 4451-X ISR Fan tray"
PID: ACS-4450-FANASSY  , VID:    , SN:

NAME: "module 0", DESCR: "Cisco ISR4451 Built-In NIM controller"
PID: ISR4451/K9        , VID:    , SN:

NAME: "NIM subslot 0/1", DESCR: " NIM-1MFT-T1/E1 - T1/E1 Serial Module"
PID:  NIM-1MFT-T1/E1  , VID: V01, SN: FOC16254E71

NAME: "subslot 0/1 db module 0", DESCR: "PVDM4-TDM-280 Voice DSP Module"
PID: PVDM4-TDM-280     , VID: V01, SN: FOC16290GRT

NAME: "NIM subslot 0/0", DESCR: "Front Panel 4 ports Gigabitethernet Module"
PID: ISR4451-X-4x1GE   , VID: V01, SN: JAB092709EL

NAME: "module 1", DESCR: "Cisco ISR4451 Built-In SM controller"
PID: ISR4451/K9        , VID:    , SN:

NAME: "module 2", DESCR: "Cisco ISR4451 Built-In SM controller"
PID: ISR4451/K9        , VID:    , SN:

NAME: "SM subslot 2/0", DESCR: "SM-X-1T3/E3 - Clear T3/E3 Serial Module"
PID: SM-1T3/E3         , VID: V01, SN: FOC15495HSE

NAME: "module R0", DESCR: "Cisco ISR 4451-X Route Processor"
PID: ISR4451/K9        , VID: V01, SN: FOC163679GH

NAME: "module F0", DESCR: "Cisco ISR4451-X Forwarding Processor"
PID: ISR4451/K9        , VID:    , SN:
```

Note Cisco ISR 4321 does not display the serial numbers of power supply and fan tray with the **show inventory** command.

Monitoring and Troubleshooting Modules and Interfaces

Use the following commands in global configuration mode to monitor and troubleshoot the modules and interfaces:

- **show platform**
- **show platform software backplaneswitch-manager RP [active [detail]]**
- **show platform hardware backplaneswitch-manager RPactive CP statistics**
- **show platform hardware backplaneswitch-manager RP active summary**
- **show platform hardware backplaneswitch-manager [R0 [status] | RP]**
- **show diag all eeprom details**

show platform

```
Router# show platform
Chassis type: ISR4451/K9

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         ISR4451/K9          ok                    15:57:33
 0/0      ISR4451-4X1GE       ok                    15:55:24
1         ISR4451/K9          ok                    15:57:33
 1/0      SM-1T3/E3           ok                    15:55:24
2         ISR4451/K9          ok                    15:57:33
 2/0      SM-1T3/E3           ok                    15:55:24
R0        ISR4451/K9          ok, active            15:57:33
F0        ISR4451-FP          ok, active            15:57:33
P0        Unknown             ps, fail              never
P1        XXX-XXXX-XX         ok                    15:56:58
P2        ACS-4450-FANASSY    ok                    15:56:58

Slot      CPLD Version        Firmware Version
--------- ------------------- ---------------------------------------
0         12090323            15.3(01r)S            [ciscouser-ISRRO...
1         12090323            15.3(01r)S            [ciscouser-ISRRO...
2         12090323            15.3(01r)S            [ciscouser-ISRRO...
R0        12090323            15.3(01r)S            [ciscouser-ISRRO...
F0        12090323            15.3(01r)S            [ciscouser-ISRRO...
```

Table 42: show platform Field Descriptions

Field	Description
Slot	Slot number
Type	Type of module
State	Status of module
Insert Time	Time since the module has been up and running

show platform software backplaneswitch-manager RP [active [detail]]

```
Router# show platform software backplaneswitch-manager RP active detail
BSM Software Display

 module port   port type  alien type    traf type
-------------------------------------------------
       0/1/0        NGIO       TRUNK         NGIO
       0/1/1        NGIO       TRUNK         NGIO
```

```
0/2/0        NGIO       TRUNK        NGIO
0/2/1        NGIO       TRUNK        NGIO
0/3/0        NGIO       TRUNK        NGIO
0/3/1       ALIEN       TRUNK        NGIO
0/4/0        NGIO       TRUNK        NGIO
0/4/1        NGIO       TRUNK        NGIO
1/0/0        NGIO       TRUNK        NGIO
1/0/1        NGIO       TRUNK        NGIO
2/0/0        NGIO       TRUNK        NGIO
2/0/1        NGIO       TRUNK        NGIO
```

show platform hardware backplaneswitch-manager RPactive CP statistics

```
Router# show platform hardware backplaneswitch-manager RP active CP statistics
Broadcom 1G port(e.g:  NIM, NGSM, CP) status:
                         Rx pkts         Rx Bytes          Tx Pkts         Tx Bytes
-----------------------------------------------------------------------------------------
All                         6242         9361008             6241           403209
  =64                         72                             6178
  65~127                       4                               60
  128~255                      0                                0
  256~511                      3                                1
  512~1023                     0                                2
  1024~1518                 6163                                0
  1519~2047                    0                                0
  2048~4095                    0                                0
  4096~9216                    0                                0
Good                        6242                             6241
  CoS 0                                                         0                0
  CoS 1                                                         0                0
  CoS 2                                                         0                0
  CoS 3                                                      6241           403209
  CoS 4                                                         0                0
  CoS 5                                                         0                0
  CoS 6                                                         0                0
  CoS 7                                                         0                0
  Unicast                   6241                             6235
  Multicast                    1                                0
  Broadcast                    0                                6
  Control                      0                                0
  VLAN                         0                                0
Errored
  FCS                          0                                0
  Runts                        0               0
  Undersize                    0
  Ether len                    0
  Fragment                     0                                0
  Jabber                       0                                0
  MTU                          0
Drops
  CoS 0                                                         0                0
  CoS 1                                                         0                0
  CoS 2                                                         0                0
  CoS 3                                                         0                0
  CoS 4                                                         0                0
  CoS 5                                                         0                0
  CoS 6                                                         0                0
  CoS 7                                                         0                0
  STP                          0
  backpress                    0
  congest                      0               0
  purge/cell                   0
  no destination               1
Pause                          0                                0
```

show platform hardware backplaneswitch-manager RP active summary

```
Router# show platform hardware backplaneswitch-manager RP active summary
  slot     bay          port          InBytes            InPkts         OutBytes  OutPkts

------------------------------------------------------------------------------------------
     0       0            CP              242                 0                  0   0
     1       0           GE1                0                 0                  0   0
     1       0           GE0                0                 0                  0   0
     2       0           GE1                0                 0                  0   0
     2       0           GE0                0                 0                  0   0
     0       1           GE1                0                 0                  0   0
     0       1           GE0                0                 0                  0   0
     0       2           GE1                0                 0                  0   0
     0       2           GE0                0                 0                  0   0
     0       3           GE1                0                 0                  0   0
     0       3           GE0                0                 0                  0   0
     0       4           GE1                0                 0                  0   0
     0       4           GE0                0                 0                  0   0
     0       0           FFP                0                 0                  0   0
```

show platform hardware backplaneswitch-manager [R0 [status] | RP]

```
Router# show platform hardware backplaneswitch-manager R0 status
slot  bay   port   enable   link status   speed(Mbps)   duplex   autoneg   pause_tx
pause_rx   mtu
------------------------------------------------------------------------------------------
0      0     CP     True      Up           1000          Full    ENABLED    ENABLED
ENABLED   10240
1      0     GE1    True      Up           1000          Full    DISABLED   ENABLED
ENABLED   10240
1      0     GE0    True      Up           1000          Full    DISABLED   ENABLED
ENABLED   10240
2      0     GE1    True      Up           1000          Full    DISABLED   ENABLED
ENABLED   10240
2      0     GE0    True      Up           1000          Full    DISABLED   ENABLED
ENABLED   10240
0      1     GE1    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      1     GE0    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      2     GE1    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      2     GE0    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      3     GE1    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      3     GE0    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      4     GE1    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      4     GE0    True     Down          1000          Full    DISABLED   ENABLED
ENABLED   10240
0      0     FFP    True       Up         10000          Full    ENABLED    DISABLED
DISABLED  10240
slot  bay   port            mac    vid   modid     flags - Layer 2
----------------------------------------------------------------------------
0      0    FFP  2c54.2dd2.661b   2351       1             0x20
0      0    FFP  2c54.2dd2.661b   2352       1             0x20
0      0     CP  2c54.2dd2.661e   2351       0             0xC60
0      0     CP  2c54.2dd2.661e   2352       0             0x20
1      0    GE0  58bf.ea3a.00f6   2350       0             0x460
0      0    FFP  2c54.2dd2.661b   2350       1             0x20
1      0    GE0  58bf.ea3a.00f6   2352       0             0x20
```

```
0      0     CP  2c54.2dd2.661e   2350       0              0x20
1      0    GE0  58bf.ea3a.00f6   2351       0              0xC60
Port block masks: rows=from port, columns=to port, u=unknown unicast, m=unknown multicast,
 b=broadcast, A=all

          CP    FFP  1/0/1  1/0/0  2/0/1  2/0/0  0/1/1  0/1/0  0/2/1  0/2/0  0/3/1  0/3/0
  0/4/1 0/4/0 drops
------------------------------------------------------------------------------------------
CP         -      A     um     um     um     um     um     um     um     um     um     um
     um     um     1
FFP        A      -      -      -      -      -      -      -      -      -      -      -
      -      -     0
1/0/1      um    umb     -    umb    umb    umb    umb    umb    umb    umb    umb    umb
    umb    umb     0
1/0/0      um    umb   umb     -    umb    umb    umb    umb    umb    umb    umb    umb
    umb    umb     6
2/0/1      um    umb   umb    umb     -     umb    umb    umb    umb    umb    umb    umb
    umb    umb     0
2/0/0      um    umb   umb    umb    umb     -     umb    umb    umb    umb    umb    umb
    umb    umb     6
0/1/1      um    umb   umb    umb    umb    umb     -     umb    umb    umb    umb    umb
    umb    umb     0
0/1/0      um    umb   umb    umb    umb    umb    umb     -     umb    umb    umb    umb
    umb    umb     0
0/2/1      um    umb   umb    umb    umb    umb    umb    umb     -     umb    umb    umb
    umb    umb     0
0/2/0      um    umb   umb    umb    umb    umb    umb    umb    umb     -     umb    umb
    umb    umb     0
0/3/1      um    umb   umb    umb    umb    umb    umb    umb    umb    umb     -     umb
    umb    umb     0
0/3/0      um    umb   umb    umb    umb    umb    umb    umb    umb    umb    umb     -
    umb    umb     0
0/4/1      um    umb   umb    umb    umb    umb    umb    umb    umb    umb    umb    umb
     -     umb     0
0/4/0      um    umb   umb    umb    umb    umb    umb    umb    umb    umb    umb    umb
    umb     -      0

Port VLAN membership: [untagged vlan] U=untagged T=tagged <VLAN range begin>-<VLAN range
end>

    CP [2352] U:0001-0001 T:0002-2351 U:2352-2352 T:2353-4095
   FFP [2352] T:0001-4095
 1/0/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 1/0/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 2/0/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 2/0/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/1/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/1/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/2/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/2/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/3/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/3/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/4/1 [2352] T:0002-2351 U:2352-2352 T:2353-4095
 0/4/0 [2352] T:0002-2351 U:2352-2352 T:2353-4095
```

show diag all eeprom details

```
Router# show diag all eeprom details
MIDPLANE EEPROM data:

        EEPROM version           : 4
        Compatible Type          : 0xFF
        PCB Serial Number        : FOC15520B7L
        Controller Type          : 1902
```

```
        Hardware Revision        : 1.0
        PCB Part Number          : 73-13854-02
        Top Assy. Part Number    : 800-36894-01
        Board Revision           : 05
        Deviation Number         : 123968
        Fab Version              : 02
        Product Identifier (PID) : ISR4451/K9
        Version Identifier (VID) : V01
        CLEI Code                : TDBTDBTDBT
        Processor type           : D0
        Chassis Serial Number    : FGL1601129D
        Chassis MAC Address      : 30f7.0d53.c7e0
        MAC Address block size   : 144
        Manufacturing Test Data  : 00 00 00 00 00 00 00 00
        Asset ID                 : P1B-R2C
Power/Fan Module P0 EEPROM data:

        EEPROM version           : 4
        Compatible Type          : 0xFF
        Controller Type          : 1509
        Unknown Field (type 00DF): 1.85.1.236.1
        Deviation Number         : 0
        PCB Serial Number        : DCA1547X037
        RMA Test History         : 00
        RMA Number               : 0-0-0-0
        RMA History              : 00
        Version Identifier (VID) : XXX
        Product Identifier (PID) : XXX-XXXX-XX
        CLEI Code                : 0000000000
        Environment Monitor Data : 41 01 C2 42 00 05 F8 00
                                   50 01 F4 1B 58 03 E8 1F
                                   4A 05 DC 21 34 07 D0 21
                                   FC 09 C4 22 60 0B B8 22
                                   92 0D AC 22 D8 0F A0 22
                                   F8 11 94 22 F6 13 88 23
                                   3C 15 7C 23 28 17 70 23
                                   00 19 64 22 D8 1B 58 22
                                   C4 1D 4C 22 BA 1F 40 22
                                   A6 21 34 22 9C 23 28 22
                                   92 25 1C 22 88 27 10 22
                                   60
        Board Revision           : P0
Power/Fan Module P1 EEPROM data is not initialized

Power/Fan Module P2 EEPROM data is not initialized

Slot R0 EEPROM data:

        EEPROM version           : 4
        Compatible Type          : 0xFF
        PCB Serial Number        : FOC15520B7L
        Controller Type          : 1902
        Hardware Revision        : 1.0
        PCB Part Number          : 73-13854-02
        Top Assy. Part Number    : 800-36894-01
        Board Revision           : 05
        Deviation Number         : 123968
        Fab Version              : 02
        Product Identifier (PID) : ISR4451/K9
        Version Identifier (VID) : V01
        CLEI Code                : TDBTDBTDBT
        Processor type           : D0
        Chassis Serial Number    : FGL1601129D
        Chassis MAC Address      : 30f7.0d53.c7e0
```

```
        MAC Address block size   : 144
        Manufacturing Test Data  : 00 00 00 00 00 00 00 00
        Asset ID                 : P1B-R2C
        Asset ID                 :
Slot F0 EEPROM data:

        EEPROM version           : 4
        Compatible Type          : 0xFF
        Controller Type          : 3567
        Hardware Revision        : 4.1
        PCB Part Number          : 73-12387-01
        MAC Address block size   : 15
        Chassis MAC Address      : aabb.ccdd.eeff
        Product Identifier (PID) : ISR4451-FP
        Version Identifier (VID) : V00
        PCB Serial Number        : FP123456789
        Asset ID                 :
Slot 0 EEPROM data:

        EEPROM version           : 4
        Compatible Type          : 0xFF
        Controller Type          : 1612
        Hardware Revision        : 4.1
        PCB Part Number          : 73-12387-01
        MAC Address block size   : 15
        Chassis MAC Address      : aabb.ccdd.eeff
        Product Identifier (PID) : ISR4451-NGSM
        Version Identifier (VID) : V00
        PCB Serial Number        : NGSM1234567
        Asset ID                 :
Slot 1 EEPROM data:

        EEPROM version           : 4
        Compatible Type          : 0xFF
        Controller Type          : 1612
        Hardware Revision        : 4.1
        PCB Part Number          : 73-12387-01
        MAC Address block size   : 15
        Chassis MAC Address      : aabb.ccdd.eeff
        Product Identifier (PID) : ISR4451-NGSM
        Version Identifier (VID) : V00
        PCB Serial Number        : NGSM1234567
        Asset ID                 :
Slot 2 EEPROM data:

        EEPROM version           : 4
        Compatible Type          : 0xFF
        Controller Type          : 1612
        Hardware Revision        : 4.1
        PCB Part Number          : 73-12387-01
        MAC Address block size   : 15
        Chassis MAC Address      : aabb.ccdd.eeff
        Product Identifier (PID) : ISR4451-NGSM
        Version Identifier (VID) : V00
        PCB Serial Number        : NGSM1234567
        Asset ID                 :
SPA EEPROM data for subslot 0/0:

        EEPROM version           : 5
        Compatible Type          : 0xFF
        Controller Type          : 1902
        Hardware Revision        : 2.2
        Boot Timeout             : 400 msecs
        PCB Serial Number        : JAB092709EL
```

```
        PCB Part Number          : 73-8700-01
        PCB Revision             : A0
        Fab Version              : 01
        RMA Test History         : 00
        RMA Number               : 0-0-0-0
        RMA History              : 00
        Deviation Number         : 78409
        Product Identifier (PID) : ISR4451-4X1GE
        Version Identifier (VID) : V01
        Top Assy. Part Number    : 68-2236-01
        Top Assy. Revision       : A0
        IDPROM Format Revision   : 36
        System Clock Frequency   : 00 00 00 00 00 00 00 00
                                   00 00 00 00 00 00 00 00
                                   00 00 00 00 00 00
        CLEI Code                : CNUIAHSAAA
        Base MAC Address         : 00 00 00 00 00 00
        MAC Address block size   : 0
        Manufacturing Test Data  : 00 00 00 00 00 00 00 00
        Field Diagnostics Data   : 00 00 00 00 00 00 00 00
        Calibration Data         : Minimum: 0 dBmV, Maximum: 0 dBmV
              Calibration values :
        Power Consumption        : 13100 mWatts (Maximum)
        Environment Monitor Data : 03 30 0C E4 46 32 09 C4
                                   46 32 05 DC 46 32 05 DC
                                   46 32 00 00 00 00 00 00
                                   00 00 00 00 00 00 00 00
                                   00 00 00 00 00 00 00 00
                                   00 00 00 00 00 00 00 00
                                   00 00 FE 02 F9 6E
        Processor Label          : 00 00 00 00 00 00 00
        Platform features        : 00 00 00 00 00 00 00 00
                                   00 00 00 00 00 00 00 00
                                   00 00 00 00 00 00 00 00
                                   00 00 00 00 00 00 00
        Asset ID                 :
        Asset Alias              :
SPA EEPROM data for subslot 0/1 is not available

SPA EEPROM data for subslot 0/2 is not available

SPA EEPROM data for subslot 0/3 is not available

SPA EEPROM data for subslot 0/4 is not available

SPA EEPROM data for subslot 1/0 is not available

SPA EEPROM data for subslot 1/1 is not available

SPA EEPROM data for subslot 1/2 is not available

SPA EEPROM data for subslot 1/3 is not available

SPA EEPROM data for subslot 1/4 is not available

SPA EEPROM data for subslot 2/0 is not available

SPA EEPROM data for subslot 2/1 is not available

SPA EEPROM data for subslot 2/2 is not available

SPA EEPROM data for subslot 2/3 is not available

SPA EEPROM data for subslot 2/4 is not available
```

Configuration Examples

This section provides examples of deactivating and activating modules.

Deactivating a Module Configuration: Example

You can deactivate a module to perform OIR of that module. The following example shows how to deactivate a module (and its interfaces) and remove power to the module. In this example, the module is installed in subslot 0 of the router.

```
Router(config)# hw-module slot 1 subslot 1/0 shutdown unpowered
```

Activating a Module Configuration: Example

You can activate a module if you have previously deactivated it. If you have not deactivated a module and its interfaces during OIR, then the module is automatically reactivated upon reactivation of the router.

The following example shows how to activate a module. In this example, the module is installed in subslot 0, located in slot 1 of the router:

```
Router(config)# hw-module slot 1 subslot 1/0 start
```

CHAPTER 24

SFP Auto-Detect and Auto-Failover

Cisco 4000 Series Integrated Services Routers (ISRs) provide a Front Panel Gigabit Ethernet (FPGE) port that supports copper and fiber concurrent connections. Media can be configured for failover redundancy when the network goes down. This feature is supported only on Cisco ISR platforms.

This chapter includes this section:

Enabling Auto-Detect

When the media-type is not configured, the Auto-Detect feature is enabled by default. The Auto-Detect feature automatically detects the media that is connected and links up. If both the media are connected, whichever media comes up first is linked. By default, the media-type on FPGE ports is set to auto-select. User can overwrite the media-type configuration to either RJ-45 or SFP using the **media-type rj45/sfp** command under the FPGE interface. The media type configuration also falls back to “Auto-select” mode when the **no media-type** command is configured. You can use the **no media-type** command in interface configuration mode to enable the Auto-Detect feature.

Configuring Auto-Detect

The Auto-Detect feature is enabled by default on the Front Panel Gige Ports. It is enabled by either configuring "media-type auto-select" or "no media-type". To configure the Auto-Detect, perform these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface gigabitethernet** {**slot** | **bay** | **port**}
3. **media-type auto-select**
4. **End**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Router# configure terminal`	
Step 2	**interface gigabitethernet** {**slot** | **bay**| **port**} **Example:** `Router(config)# interface gigabitethernet slot/port`	Enters interface configuration mode.
Step 3	**media-type auto-select** **Example:** `Router(config-if)# media-type auto-select`	Auto-select mode uses whichever connector is attached. The options are: • **rj45**—Uses RJ45 connector. • **sfp**—Uses SFP connector.
Step 4	**End** **Example:** `Router(config-if)#end`	Exits to global configuration mode.

Examples

The following example shows the default configuration and the show running configuration does not show any media type when the no media-type is selected.

```
Router(config)# show running interface gigabitethernet 0/0/0
Building configuration...

Current configuration : 71 bytes
!
interface GigabitEthernet0/0/0
 no ip address
 negotiation auto
end
```

Configuring the Primary and Secondary Media

When the router receives an indication that the primary media is down, the secondary failover media is enabled. After the switchover, the media does not switch back to primary media when the primary media is restored. You need to use either **shut** or **no shut** command or reload the module to switch the media-type back to primary(preferred) media.

To assign the primary or secondary failover media on the GE-SFP port, perform these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface gigabitethernet** {**slot** | **port**}
3. **media-type rj45 autofailover**
4. **End**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**interface gigabitethernet** {**slot** | **port**} **Example:** `Router(config)# interface gigabitethernet slot/port`	Enters interface configuration mode.
Step 3	**media-type rj45 autofailover** **Example:** `Router(config-if)# media-type rj45 autofailover`	Configures the port with rj45 as the primary media for automatic failover.
Step 4	**End** **Example:** `Router(config-if)#end`	Exits to global configuration mode.

Examples

The following example shows the primary configuration.

```
Router(config)# show running interface gigabitethernet 0/0/0
Building configuration...

Current configuration : 102 bytes
!
interface GigabitEthernet0/0/0
 no ip address
 media-type rj45 auto-failover
 negotiation auto
end
```

CHAPTER 25

Cellular IPv6 Address

This chapter provides an overview of the IPv6 addresses and describes how to configure Cellular IPv6 address on Cisco 4000 series ISRs.

This chapter includes this section:

Cellular IPv6 Address

IPv6 addresses are represented as a series of 16-bit hexadecimal fields separated by colons (:) in the format: x:x:x:x:x:x:x:x. Following are two examples of IPv6 addresses:

- 2001:CDBA:0000:0000:0000:0000:3257:9652
- 2001:CDBA::3257:9652 (zeros can be omitted)

IPv6 addresses commonly contain successive hexadecimal fields of zeros. Two colons (::) may be used to compress successive hexadecimal fields of zeros at the beginning, middle, or end of an IPv6 address (the colons represent successive hexadecimal fields of zeros). The table below lists compressed IPv6 address formats.

An IPv6 address prefix, in the format ipv6-prefix/prefix-length, can be used to represent bit-wise contiguous blocks of the entire address space. The ipv6-prefix must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. The prefix length is a decimal value that indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). For example, 2001:cdba::3257:9652 /64 is a valid IPv6 prefix.

IPv6 Unicast Routing

An IPv6 unicast address is an identifier for a single interface, on a single node. A packet that is sent to a unicast address is delivered to the interface identified by that address.

Cisco 4000 Series ISR supports the following address types:

- Link-Lock Address , on page 408
- Global Address, on page 408

Link-Lock Address

A link-local address is an IPv6 unicast address that can be automatically configured on any interface using the link-local prefix FE80::/10 (1111 1110 10) and the interface identifier in the modified EUI-64 format. An link-local address is automatically configured on the cellular interface when an IPv6 address is enabled.

After the data call is established, the link-local address on the celluar interface is updated with the host generated link-local address that consists of the link-local prefix FF80::/10 (1111 1110 10) and the auto-generated interface identifier from the USB hardware address. The figure below shows the structure of a link-local address.

Global Address

A global IPv6 unicast address is defined by a global routing prefix, a subnet ID, and an interface ID. The routing prefix is obtained from the PGW. The Interface Identifier is automatically generated from the USB hardware address using the interface identifier in the modified EUI-64 format. The USB hardware address changes after the router reloads.

Configuring Cellular IPv6 Address

To configure the cellular IPv6 address, perform these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface Cellular** {**type** | **number**}
3. ip address negotiated
4. encapsulation slip
5. load-interval***seonds***
6. dialer in-band
7. dialer idle-timeout ***seonds***
8. dialer string **string**
9. dialer-group**group-number**
10. no peer default ip address
11. ipv6 address autoconfig
12. async mode interactive
13. routing dynamic
14. **dialer-listdialer-groupprotocolprotocol-name** {**permit** |deny|**list** |*access-list-number* | *access-group* }
15. **ipv6 route** *ipv6-prefix/prefix-length 128*
16. **End**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface Cellular** {**type** | **number**} **Example:** `Router(config)# interface cellular 0/1/0`	Specifies the cellular interface.
Step 3	ip address negotiated **Example:** `Router(config-if)# ipv6 address negotiated`	Specifies that the IP address for a particular interface is dynamically obtained.
Step 4	encapsulation slip **Example:** `Router(config-if)# encapsulation slip`	Specifies Serial Line Internet Protocol (SLIP) encapsulation for an interface configured for dial-on-demand routing (DDR).
Step 5	load-interval***seonds*** **Example:** `Router(config-if)# load-interval 30`	Specifies the length of time for which data is used to compute load statistics.
Step 6	dialer in-band **Example:** `Router(config-if)# dialer in-band`	Enables DDR and configures the specified serial interface to use in-band dialing.
Step 7	dialer idle-timeout ***seonds*** **Example:** `Router(config-if)# dialer idle-timeout 0`	Specifies the dialer idle timeout period.
Step 8	dialer string **string** **Example:** `Router(config-if)# dialer string lte`	Specifies the number or string to dial.
Step 9	dialer-group**group-number** **Example:** `Router(config-if)# dialer-group 1`	Specifies the number of the dialer access group to which the specific interface belongs.
Step 10	no peer default ip address **Example:** `Router(config-if)# no peer default ip address`	Removes the default address from your configuration.
Step 11	ipv6 address autoconfig **Example:** `Router(config-if)# ipv6 address autoconfig`	Enables automatic configuration of IPv6 addresses using stateless autoconfiguration on an interface and enables IPv6 processing on the interface.
Step 12	async mode interactive **Example:** `Router(config-if)# async mode interactive`	Please provide the inputs?

	Command or Action	Purpose
Step 13	routing dynamic **Example:** `Router(config-if)#routing dynamic`	Enables the router to pass routing updates to other routers through an interface.
Step 14	**dialer-listdialer-groupprotocolprotocol-name** {**permit** |deny|**list** |*access-list-number* | *access-group* } **Example:** `Router(config)# dialer-list 1 protocol ipv6 permit`	Defines a dial-on-demand routing (DDR) dialer list for dialing by protocol or by a combination of a protocol and a previously defined access list.
Step 15	**ipv6 route** *ipv6-prefix/prefix-length 128* **Example:** `Router(config)#ipv6 route 2001:1234:1234::3/128 Cellular0/1/0`	
Step 16	**End** **Example:** `Router(config-if)#end`	Exits to global configuration mode.

Examples

The following example shows the Cellular IPv6 configuration .

```
Router(config)# interface Cellular0/0/0
ip address negotiated
encapsulation slip
load-interval 30
dialer in-band
dialer idle-timeout 0
dialer string lte
dialer-group 1
no peer default ip address
ipv6 address autoconfig
async mode interactive
routing dynamic
!
interface Cellular0/1/0
ip address negotiated
encapsulation slip
load-interval 30
dialer in-band
dialer idle-timeout 0
dialer string lte
dialer-group 1
no peer default ip address
ipv6 address autoconfig
async mode interactive
routing dynamic

dialer-list 1 protocol ipv6 permit
ipv6 route 2001:1234:1234::/64 Cellular0/1/0
ipv6 route 2001:4321:4321::5/128 Cellular0/1/1
```

CHAPTER 26

Radio Aware Routing

Radio-Aware Routing (RAR) is a mechanism that uses radios to interact with the routing protocol OSPFv3 to signal the appearance, disappearance, and link conditions of one-hop routing neighbors.

In a large mobile networks, connections to the routing neighbors are often interrupted due to distance and radio obstructions. When these signals do not reach the routing protocols, protocol timers are used to update the status of a neighbor. Routing protocols have lengthy timer, which is not recommended in mobile networks.

The RAR feature is supported on Cisco ISR G2 and G3 Series Routers, Cisco ISR 4000 Series Routers.

PPPoE Extensions is the RAR protocol supported in Cisco 4000 Series ISRs. PPPoE Extensions with Aggregate support is introduce from Cisco IOS XE Fuji 16.7. release. OSPFv3 and EIGRP are the supported routing protocols.

Benefits of Radio Aware Routing

The Radio Aware Routing feature offers the following benefits:

- Provides faster network convergence through immediate recognition of changes.
- Enables routing for failing or fading radio links.
- Allows easy routing between line-of-sight and non-line-of-sight paths.
- Provides faster convergence and optimal route selection so that delay-sensitive traffic, such as voice and video, is not disrupted
- Provides efficient radio resources and bandwidth usage.
- Reduces impact on the radio links by performing congestion control in the router.

- Allows route selection based on radio power conservation.
- Enables decoupling of the routing and radio functionalities.
- Provides simple Ethernet connection to RFC 5578, R2CP, and DLEP compliant radios.

Restrictions and Limitations

The Radio Aware Routing feature has the following restrictions and limitations:

- The DLEP and R2CP protocols are not supported in Cisco 4000 Series ISRs.
- Multicast traffic is not supported in aggregate mode.
- Cisco High Availability (HA) technology is not supported.

License Requirements

This feature is available with the AX license.

System Components

The Radio Aware Routing (RAR) feature is implemented using the MANET (Mobile adhoc network) infrastructure comprising of different components such as PPPoE, Virtual multipoint interface (VMI), QoS, routing protocol interface and RAR protocols.

Point-to-Point Protocol over Ethernet PPPoE or PPPoE

PPPoE is a well-defined communication mechanism between the client and the server. In the RAR implementation, radio takes the role of the PPPoE client and router takes the role of the PPPoE server. This allows a loose coupling of radio and router, while providing a well-defined and predictable communication mechanism.

As PPPoE is a session or a connection oriented protocol, it extends the point-to-point radio frequency (RF) link from an external radio to an IOS router.

PPPoE Extensions

PPPoE extensions are used when the router communicates with the radio. In the Cisco IOS implementation of PPPoE, each individual session is represented by virtual access interface (connectivity to a radio neighbor) on which, QoS can be applied with these PPPoE extensions.

RFC5578 provides extensions to PPPoE to support credit-based flow control and session-based real time link metrics, which are very useful for connections with variable bandwidth and limited buffering capabilities (such as radio links).

Virtual Multipoint Interface (VMI)

Though PPPoE Extensions provides the most of the setup to communicate between a router and a radio, VMI addresses the need to manage and translate events that higher layers (example, routing protocols) consume. In addition, VMI operates in the Bypass mode.

In Bypass mode, every Virtual Access Interface (VAI) representing a radio neighbor is exposed to routing protocols OSPFv3 and EIGRP, so that, the routing protocol directly communicates with the respective VAI for both unicast and multicast routing protocol traffic.

In Aggregae mode, VMI is exposed to the routing protocols (OSPF) so that the routing protocols can leverage VMI for their optimum efficiency. When the network neighbors are viewed as a collection of networks on a point-to-multipoint link with broadcast and multicast capability at VMI, VMI helps in aggregating the multiple virtual access interfaces created from PPPoE. VMI presents a single multi access layer 2 broadcast capable interface. The VMI layer handles re-directs unicast routing protocol traffic to the appropriate P2P link (Virtual-Access interface), and replicats any Multicast/Broadcast traffic that needs to flow. Since the routing protocol communicates to a single interface, the size of the topology database is reduced, without impacting the integrity of the network.

QoS Provisioning on PPPoE Extension Session

The following example describes QoS provisioning on PPPoE extension session:

```
policy-map rar_policer
 class class-default
  police 10000 2000 1000 conform-action transmit  exceed-action drop  violate-action drop
policy-map rar_shaper
 class class-default
  shape average percent 1

interface Virtual-Template2
 ip address 10.92.2.1 255.255.255.0
 no peer default ip address
 no keepalive
 service-policy input rar_policer
end
```

Example: Configuring the RAR Feature in Bypass Mode

The following example is an end-to-end configuration of RAR in the bypass mode:

Note Before you being the RAR configuration, you must first configure the **subscriber authorization enable** command to bring up the RAR session. Without enbaling authorization, the Point-to-Point protocol does not recognize this as a RAR session and may not tag *manet_radio* in presentation of a PPPoE Active Discovery Initiate (PADI). By default, bypass mode does not appears in the configuration. It appears only if the mode is configured as bypass.

Configure a Service for RAR

```
policy-map type service rar-lab
 pppoe service manet_radio //note: Enter the pppoe service policy name as manet_radio
!
```

Configure Broadband

```
bba-group pppoe VMI2
 virtual-template 2
service profile rar-lab
!
interface GigabitEthernet0/0/0
 description Connected to Client1
  negotiation auto
  pppoe enable group VMI2
!
```

Configure a Service for RAR

```
policy-map type service rar-lab
 pppoe service manet_radio //note: Enter the pppoe service policy name as manet_radio
!
```

Configuration in Bypass Mode

- IP Address Configured under Virtual-Template Explicitly

```
interface Virtual-Template2
ip address 192.0.2.3 255.255.255.0
no ip redirects
peer default ip address pool PPPoEpool2
ipv6 enable
ospfv3 1 network manet
ospfv3 1 ipv4 area 0
ospfv3 1 ipv6 area 0
no keepalive
service-policy input rar_policer Or/And
service-policy output rar_shaper
```

- VMI Unnumbered Configured under Virtual Template

```
interface Virtual-Template2
ip unnumbered vmi2
no ip redirects
peer default ip address pool PPPoEpool2
ipv6 enable
ospfv3 1 network manet
ospfv3 1 ipv4 area 0
ospfv3 1 ipv6 area 0
no keepalive
service-policy input rar_policer Or/And
service-policy output rar_shaper
```

Configure the Virtual Multipoint Interface in Bypass Mode

```
interface vmi2 //configure the virtual multi interface
ip address 192.0.2.1 255.255.255.0
```

```
physical-interface GigabitEthernet0/0/0
mode bypass

interface vmi3//configure the virtual multi interface
 ip address 192.0.2.3 255.255.255.0
 physical-interface GigabitEthernet0/0/1
mode bypass
```

Configure OSPF Routing

```
router ospfv3 1
 router-id 192.0.2.1
!
 address-family ipv4 unicast
  redistribute connected metric 1 metric-type 1
  log-adjacency-changes
 exit-address-family
 !
 address-family ipv6 unicast
  redistribute connected metric-type 1
  log-adjacency-changes
 exit-address-family
!
ip local pool PPPoEpool2 198.51.100.1 198.51.100.254
```

Example: Configuring the RAR Feature in Aggregate Mode

The following example is an end-to-end configuration of RAR in the aggregate mode:

Note Before you being the RAR configuration, you must first configure the **subscriber authorization enable** command to bring up the RAR session. Without enabling authorization, the Point-to-Point protocol does not recognize this as a RAR session and may not tag *manet_radio* in PADI.

Configure a Service for RAR

```
policy-map type service rar-lab
 pppoe service manet_radio //note: Enter the pppoe service policy name as manet_radio
!
```

Configure Broadband

```
bba-group pppoe VMI2
 virtual-template 2
service profile rar-lab

!
interface GigabitEthernet0/0/0
 description Connected to Client1
  negotiation auto
  pppoe enable group VMI2

!
```

Configure a Service for RAR

```
policy-map type service rar-lab
 pppoe service manet_radio //note: Enter the pppoe service policy name as manet_radio
!
```

Configuration in Aggregate Mode

```
interface Virtual-Template2
ip unnumbered vmi2
no ip redirects
no peer default ip address
ipv6 enable
no keepalive
service-policy input rar_policer Or/And
service-policy output rar_shaper
```

Configure the Virtual Multipoint Interface in Aggregate Mode

```
interface vmi2 //configure the virtual multi interface
ip address 192.0.2.1 255.255.255.0
physical-interface GigabitEthernet0/0/0
mode aggregate
```

```
interface vmi3//configure the virtual multi interface
 ip address 192.0.2.3 255.255.255.0
 no ip redirects
 no ip split-horizon eigrp 1
 physical-interface GigabitEthernet0/0/1
mode aggregate
```

Configure OSPF Routing

```
router ospfv3 1
 router-id 192.0.2.1
!
 address-family ipv4 unicast
  redistribute connected metric 1 metric-type 1
  log-adjacency-changes
 exit-address-family
 !
 address-family ipv6 unicast
  redistribute connected metric-type 1
  log-adjacency-changes
 exit-address-family
!
ip local pool PPPoEpool2 198.51.100.1 198.51.100.254
ip local pool PPPoEpool3 203.0.113.1 203.0.113.254
```

Verifying RAR Session Details

To retrieve RAR session details, use the following show commands:

```
Router#show pppoe session packets all
Total PPPoE sessions 2
```

```
session id: 9
local MAC address: 006b.f10e.a5e0, remote MAC address: 0050.56bc.424a
virtual access interface: Vi2.1, outgoing interface: Gi0/0/0
    1646 packets sent, 2439363 received
    176216 bytes sent, 117250290 received

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 32928     PADG Timer index: 0
 PADG last rcvd Seq Num: 17313
 PADG last nonzero Seq Num: 17306
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 33308  rcvd: 17313
 PADC xmit: 17313  rcvd: 19709
 In-band credit pkt xmit: 7 rcvd: 2434422
 Last credit packet snapshot
  PADG xmit: seq_num = 32928, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 32928, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 17313, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 17313, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 61, bcn = 65533
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 0


session id: 10
local MAC address: 006b.f10e.a5e1, remote MAC address: 0050.56bc.7dcb
virtual access interface: Vi2.2, outgoing interface: Gi0/0/1
    1389302 packets sent, 1852 received
    77869522 bytes sent, 142156 received

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 18787     PADG Timer index: 0
 PADG last rcvd Seq Num: 18784
 PADG last nonzero Seq Num: 18768
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 18787  rcvd: 18784
 PADC xmit: 18784  rcvd: 18787
 In-band credit pkt xmit: 1387764 rcvd: 956
 Last credit packet snapshot
  PADG xmit: seq_num = 18787, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 18787, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 18784, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 18784, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 0, bcn = 64222
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 1


Router#show pppoe session packets
Total PPPoE sessions 2

SID     Pkts-In         Pkts-Out        Bytes-In        Bytes-Out
```

```
9       2439391         1651            117252098       176714
10      1858            1389306         142580          77869914


Router#show vmi counters
Interface vmi2: - Last Clear Time =

Input Counts:
  Process Enqueue      =          0 (VMI)
  Fastswitch           =          0
  VMI Punt Drop:
       Queue Full      =          0

Output Counts:
  Transmit:
       VMI Process DQ  =       4280
       Fastswitch VA   =          0
       Fastswitch VMI  =          0
  Drops:
       Total           =          0
       QOS Error       =          0
       VMI State Error =          0
       Mcast NBR Error =          0
       Ucast NBR Error =          0
Interface vmi3: - Last Clear Time =

Input Counts:
  Process Enqueue      =          0 (VMI)
  Fastswitch           =          0
  VMI Punt Drop:
       Queue Full      =          0

Output Counts:
  Transmit:
       VMI Process DQ  =       2956
       Fastswitch VA   =          0
       Fastswitch VMI  =          0
  Drops:
       Total           =          0
       QOS Error       =          0
       VMI State Error =          0
       Mcast NBR Error =          0
       Ucast NBR Error =          0
Interface vmi4: - Last Clear Time =

Input Counts:
  Process Enqueue      =          0 (VMI)
  Fastswitch           =          0
  VMI Punt Drop:
       Queue Full      =          0

Output Counts:
  Transmit:
       VMI Process DQ  =          0
       Fastswitch VA   =          0
       Fastswitch VMI  =          0
  Drops:
       Total           =          0
       QOS Error       =          0
       VMI State Error =          0
       Mcast NBR Error =          0
       Ucast NBR Error =          0
Router#
```

```
Router#show vmi neighbor details
1 vmi2 Neighbors
      1 vmi3 Neighbors
      0 vmi4 Neighbors
      2 Total Neighbors

vmi2   IPV6 Address=FE80::21E:E6FF:FE43:F500
       IPV6 Global Addr=::
       IPV4 Address=192.0.2.2, Uptime=05:15:01
       Output pkts=89, Input pkts=0
       No Session Metrics have been received for this neighbor.
       Transport PPPoE, Session ID=9
       INTERFACE STATS:
          VMI Interface=vmi2,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          V-Access intf=Virtual-Access2.1,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          Physical intf=GigabitEthernet0/0/0,
             Input qcount=0, drops=0, Output qcount=0, drops=0

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 33038     PADG Timer index: 0
 PADG last rcvd Seq Num: 17423
 PADG last nonzero Seq Num: 17420
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 33418  rcvd: 17423
 PADC xmit: 17423  rcvd: 19819
 In-band credit pkt xmit: 7 rcvd: 2434446
 Last credit packet snapshot
  PADG xmit: seq_num = 33038, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 33038, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 17423, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 17423, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 61, bcn = 65533
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 0


vmi3   IPV6 Address=FE80::21E:7AFF:FE68:6100
       IPV6 Global Addr=::
       IPV4 Address=192.0.2.4, Uptime=05:14:55
       Output pkts=6, Input pkts=0
       METRIC DATA: Total rcvd=1, Avg arrival rate (ms)=0
          CURRENT: MDR=128000 bps, CDR=128000 bps
                   Lat=0 ms, Res=100, RLQ=100, load=0
          MDR      Max=128000 bps, Min=128000 bps, Avg=128000 bps
          CDR      Max=128000 bps, Min=128000 bps, Avg=128000 bps
          Latency  Max=0, Min=0, Avg=0 (ms)
          Resource Max=100%, Min=100%, Avg=100%
          RLQ      Max=100, Min=100, Avg=100
          Load     Max=0%, Min=0%, Avg=0%
       Transport PPPoE, Session ID=10
       INTERFACE STATS:
          VMI Interface=vmi3,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          V-Access intf=Virtual-Access2.2,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          Physical intf=GigabitEthernet0/0/1,
             Input qcount=0, drops=0, Output qcount=0, drops=0
```

```
PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 18896     PADG Timer index: 0
 PADG last rcvd Seq Num: 18894
 PADG last nonzero Seq Num: 18884
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 18896  rcvd: 18894
 PADC xmit: 18894  rcvd: 18896
 In-band credit pkt xmit: 1387764 rcvd: 961
 Last credit packet snapshot
  PADG xmit: seq_num = 18896, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 18896, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 18894, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 18894, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 0, bcn = 64222
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 1


Router#show vmi neighbor details vmi 2
              1 vmi2 Neighbors

vmi2   IPV6 Address=FE80::21E:E6FF:FE43:F500
       IPV6 Global Addr=::
       IPV4 Address=192.0.2.2, Uptime=05:16:03
       Output pkts=89, Input pkts=0
       No Session Metrics have been received for this neighbor.
       Transport PPPoE, Session ID=9
       INTERFACE STATS:
          VMI Interface=vmi2,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          V-Access intf=Virtual-Access2.1,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          Physical intf=GigabitEthernet0/0/0,
             Input qcount=0, drops=0, Output qcount=0, drops=0

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 33100     PADG Timer index: 0
 PADG last rcvd Seq Num: 17485
 PADG last nonzero Seq Num: 17449
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 33480  rcvd: 17485
 PADC xmit: 17485  rcvd: 19881
 In-band credit pkt xmit: 7 rcvd: 2434460
 Last credit packet snapshot
  PADG xmit: seq_num = 33100, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 33100, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 17485, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 17485, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 61, bcn = 65533
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 0
```

```
Router#show platform hardware qfp active feature ess session
Current number sessions: 2
Current number TC flow: 0
Feature Type: A=Accounting D=Policing(DRL) F=FFR M=DSCP Marking L=L4redirect P=Portbundle
T=TC

   Session    Type      Segment1      SegType1      Segment2      SegType2 Feature Other
------------------------------------------------------------------------------------------
        21     PPP 0x0000001500001022    PPPOE 0x0000001500002023    LTERM -------
        24     PPP 0x0000001800003026    PPPOE 0x0000001800004027    LTERM -------


Router#show platform software subscriber pppoe_fctl evsi 21
PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 33215     PADG Timer index: 0
 PADG last rcvd Seq Num: 17600
 PADG last nonzero Seq Num: 17554
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 33595  rcvd: 17600
 PADC xmit: 17600  rcvd: 19996
 In-band credit pkt xmit: 7 rcvd: 2434485
 Last credit packet snapshot
  PADG xmit: seq_num = 33215, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 33215, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 17600, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 17600, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 61, bcn = 65533
  In-band credit pkt rcvd: fcn = 0, bcn = 65534

BQS buffer statistics
 Current packets in BQS buffer: 0
 Total en-queue packets: 0 de-queue packets: 0
 Total dropped packets: 0

Internal flags: 0x0



Router#show platform hardware qfp active feature ess session id 21
Session ID: 21

  EVSI type: PPP
  SIP Segment ID: 0x1500001022
  SIP Segment type: PPPOE
  FSP Segment ID: 0x1500002023
  FSP Segment type: LTERM
  QFP if handle: 16
  QFP interface name: EVSI21
  SIP TX Seq num: 0
  SIP RX Seq num: 0
  FSP TX Seq num: 0
  FSP RX Seq num: 0
  Condition Debug: 0x00000000
    session


Router#show ospfv3 neighbor
```

```
          OSPFv3 1 address-family ipv4 (router-id 10.3.3.3)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
192.0.2.1        0    FULL/  -        00:01:32    19              Virtual-Access2.1

          OSPFv3 1 address-family ipv6 (router-id 10.3.3.3)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
192.0.2.1        0    FULL/  -        00:01:52    19              Virtual-Access2.1
Router#
```

```
Router#sh ip route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
a - application route
+ - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C        10.90.90.0/24 is directly connected, Virtual-Access2.1
O        10.90.90.4/32 [110/1] via 192.0.2.4, 00:00:03, Virtual-Access2.1
L        10.90.90.5/32 is directly connected, Virtual-Access2.1
      10.92.90.0/32 is subnetted, 1 subnets
C        10.92.2.21 is directly connected, Virtual-Access2.1
```

Troubleshooting Radio Aware Routing

To troubleshoot the RAR, use the following debug commands:

- **debug pppoe errors**
- **debug pppoe events**
- **debug ppp error**
- **debug vmi error**
- **debug vmi neighbor**
- **debug vmi packet**
- **debug vmi pppoe**
- **debug vmi registries**
- **debug vmi multicast**
- **debug vtemplate cloning**
- **debug vtemplate event**
- **debug vtemplate error**

- **debug plat hard qfp ac feature subscriber datapath pppoe detail**

CHAPTER 27

Session Initiation Protocol Triggered VPN

Session Initiation Protocol Triggered VPN (SIP-Triggered VPN or VPN-SIP) is a service offered by service providers where a VPN is set up using Session Initiation Protocol (SIP) for on-demand media or application sharing between peers. The VPN-SIP feature defines the process in which two SIP user agents resolve each other's IP addresses, exchange the fingerprints of their self-signed certificates, third-party certificates, or pre-shared key securely, and agree to establish an IPsec-based VPN.

Service providers offer the VPN-SIP service to their customers that have SIP-based services such as bank ATMs or branches. This VPN-SIP service replaces an ISDN connection for backup network functionality. If the primary broadband service link goes down, these bank ATMs or branches connect to their central headend or data centres through the VPN-SIP service.

The SIP server of the service provider, which coordinates the VPN-SIP service, is also used for billing of the service based on the time the service is used.

Information about VPN-SIP

Components for VPN-SIP Solution

VPN-SIP uses IPSec Static Virtual Tunnel Interface (SVTI). IPSec SVTI stays in active (UP) state even when there is no IPSec security association (SA) established between the tunnel interface and the SVTI peer.

The following are three components for the VPN-SIP Solution:

- SIP
- VPN-SIP

- Crypto (IP Security (IPsec), Internet Key Exchange (IKE), Tunnel Protection (TP), Public Key Infrastructure (PKI) modules within crypto)

Sesssion Initiation Protocol

SIP is used as a name resolution mechanism to initiate an IKE session. VPN-SIP uses SIP service to establish a VPN connection to a home or a small business router that does not have a fixed IP address. This connection is achieved using self-signed certificates or pre-shared keys. SIP negotiates the use of IKE for media sessions in the Session Description Protocol (SDP) offer-and-answer model.

SIP is statically configured. One tunnel interface must be configured for each remote SIP number.

SIP also provides billing capabilities for service providers to charge customers based on the SIP number, for using the VPN-SIP service. Billing based on SIP numbers happens in the service provider network and is independent of the end devices like Cisco VPN-SIP routers.

VPN-SIP Solution

VPN-SIP is the central block that coordinates between SIP and Crypto modules, and provides an abstraction between them.

When traffic destined to a remote network behind a SIP number is routed to the tunnel interface, the IPSec control plane gets a trigger from packet switching path as there is no IPSEC SA configured to that peer. IPsec control plane passes the trigger to VPN-SIP as the tunnel is configured for VPN-SIP.

Note Static routes for remote networks for that SIP number must be configured to point to that tunnel interface.

When the VPN-SIP service is triggered, SIP sets up the call with a SIP phone number pair. SIP also passes incoming call details to the VPN-SIP and negotiates IKE media sessions using local address and fingerprint information of the local self-signed certificate or pre-shared key. SIP also passes remote address and fingerprint information to VPN-SIP.

The VPN-SIP service listens to tunnel status updates and invokes SIP to tear down the SIP session. The VPN-SIP service also provides a means to display current and active sessions.

Feature at a glance

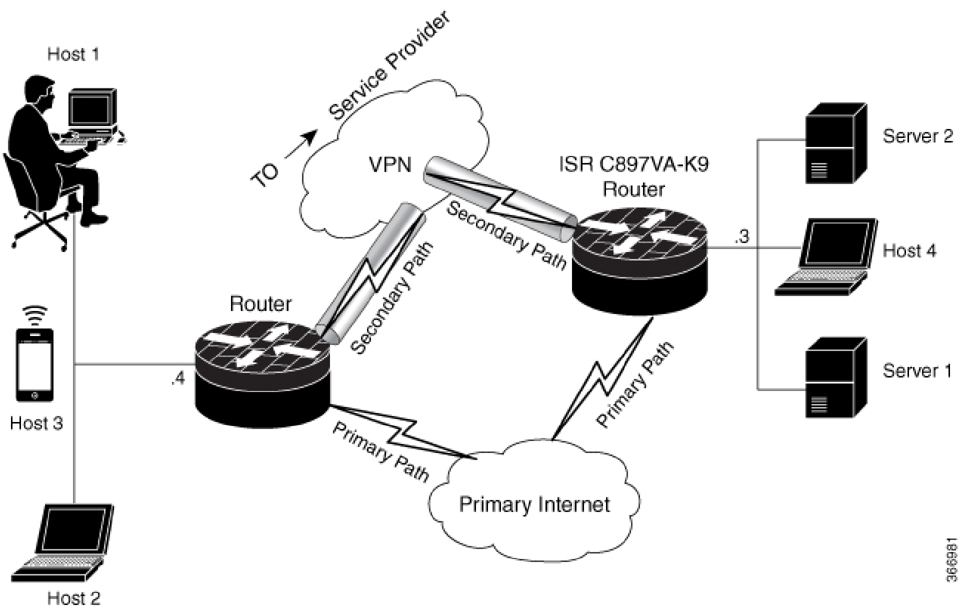
The following steps summarize how the VPN-SIP feature works:

- IP SLA monitors the primary link using route tracking. When the primary link fails IP SLA detects this failure.
- Once the primary path fails, IP SLA switches the default route to the higher metric route that is configured on the router.
- When relevant traffic tries to flow using the secondary link, SIP sends an invite message to the SIP server to obtain the VPN peer information.
- The router receives the VPN peer information (IP address, local and remote SIP numbers, IKE port, and finger print) and it establishes VPN-SIP tunnel.

- When the primary path comes back up, IP SLA detects the primary path and the route falls back to the original path. When the idle timer expires, IPSec is torn down and a SIP call is disconnected.

Following is the topology for the VPN-SIP solution:

Figure 3: VPN-SIP Topology



SIP Call Flow

The SIP call flow is divided into initiation at the local peer and call receipt at the remote peer.

At SIP Call Intitiation

When packets are routed to an SVTI interface in data plane, the SIP call must be placed to the peer SIP number to resolve its address, so that VPN tunnel can be brought up.

- When local auth-type is PSK, IKEv2 finds the matching key for a peer SIP number. The IKEv2 keyring must be configured with id_key_id type (string) as SIP number for each SIP peer. IKEv2 computes the fingerprint of the looked-up key and passes it to VPN-SIP.
- When local auth-type is a self-signed certificate or an third-party certificate, IKEv2 computes the fingerprint of the local certificate configured under the IKEv2 profile and passes it to the VPN-SIP

The VPN-SIP module interacts with SIP to setup SIP call to the peer. When the call is successful, VPN-SIP sets the tunnel destination of SVTI to the resolved IP address, requesting SVTI to initiate the VPN tunnel.

Note When a wildcard key is required, use the authentication local pre-share key command and the authentication remote pre-share key command in IKEv2 profile.

When SIP call is received at the remote peer

When a SIP call is received from a peer, following interactions occur between various crypto modules:

- The Tunnel Protection helps VPN-SIP module to set tunnel destination address.
- IKEv2 returns local auth-type (PSK or PKI) and local fingerprint to the VPN-SIP module. When local auth-type is PSK, IKEv2 finds a matching key for a corresponding SIP number.

Note IKEv2 only knows peer by its SIP number.

During the SIP call negotiation between peers, each peer must select a unique local IKEv2 port number to be exchanged over the SDP. To support different port numbers for each session, the VPN-SIP module programmatically configures IP Port Address Translation (PAT) to translate between IKEv2 port (4500) and the port number exchanged over SDP. For the translation to work IP NAT must be configured on secondary link and the loopback interface configured as the VPN-SIP tunnel source. The lifetime of the translation is limited to the lifetime of the VPN-SIP session.

SDP Offer and Answer

Following is the sample for SDP offer and answer that is negotiated in the SIP call as defined in RFC 6193:

```
offer SDP
      ...
      m=application 50001 udp ike-esp-udpencap
      c=IN IP4 10.6.6.49
      a=ike-setup:active
      a=fingerprint:SHA-1 \
      b=AS:512
      4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
      ...

   answer SDP
      ...
      m=application 50002 udp ike-esp-udpencap
      c=IN IP4 10.6.6.50
      a=ike-setup:passive
      a=fingerprint:SHA-1 \
      b=AS:512
      D2:9F:6F:1E:CD:D3:09:E8:70:65:1A:51:7C:9D:30:4F:21:E4:4A:8E
```

As part of the SDP negotiation, both peers negotiate the maximum bandwidth rate for the VPN-SIP session using the b=AS :number SDP attribute. if the peers mention different bandwidth numbers in their SDP, both of them should honor the minimum value as the maximum bandwidth. If b=AS :number SDP attribute is missing in the offer or answer, the SIP call is not successfully set up.

The negotiated maximum bandwidth is applied on the SVTI tunnel interface through the programmatically configured QoS policy in the output direction. The programmatically configured QoS policy is not applied and session fails, if there is a pre-existing statically configured policy.

Once SIP call is complete and address of the peer is resolved, VPN-SIP sets tunnel destination of SVTI and sends a request to initiate tunnel.

IKEv2 Negotiation

Following is the process for IKEv2 Security Session (SA) negotiation:

- Before starting the session, IKEv2 checks with VPN-SIP if the session is a VPN-SIP session.
- If it's a VPN-SIP session and local auth-type is PSK, IKEv2 looks up the PSK key pair using SIP number of the peer instead of IP address of the peer.
- For validating self-signed certificate, IKEv2 checks if the certificate is self-signed and validates the certificate.
    - In addition to existing AUTH payload validation as part of IKEv2 protocol, IKEv2 calculates hash of the received certificate or looked-up PSK and compares with the fingerprint from SIP negotiation that IKEv2 queries from VPN-SIP module. Only if the fingerprint matches, IKEv2 considers authentication of peer is valid. If not, IKEv2 declares that peer has failed to authenticate and fails the VPN session.

VPN-SIP solution depends on IPSEC idle timer to detect that traffic is no longer routed over the backup VPN. The idle-time configuration under the IPSec Profile is mandatory for session to be disconnected when there is no traffic. 120 seconds is the recommended time.

VPN-SIP and SIP coordinate to tear down SIP call.

When IPsec idle time expires the VPN-SIP module informs the IKEv2 to bring down the IPsec tunnel. VPN-SIP requests the SIP module to disconnect the SIP call, without waiting for confirmation from the IKEv2.

When SIP call disconnect is received from the peer, VPN-SIP module informs the IKEv2 to bring down the IPsec tunnel, and acknowledges to SIP to tear down the SIP call.

Supported Platforms

The VPN-SIP feature is supported on the following platforms:

Prerequisites for VPN-SIP

- Security K9 license must be enabled on the router.
- The routers must have a minimum memory of 1 GB.
- For the SIP register request of the SIP User Agent to succeed, the SIP registrar must be available to the VPN-SIP routers.
- The DHCP server must support option 120 and 125 to obtain the SIP server address, which is needed for registration and establishing the SIP session.
- Proper routing configurations must be completed to ensure backup WAN path is used when primary path is down.
- Maximum Transmission Unit (MTU) of the tunnel interface must be less than the MTU of the secondary WAN interface.
- When self-signed or third-party certificates are used for IKEv2 authentication, configure IKEv2 fragmentation on the VPN-SIP router to avoid fragmentation at the IP layer.
- NAT SIP ALG must be disabled.
- Caller ID notification service must be configured in the network.

Restrictions for VPN-SIP

- VPN-SIP and CUBE/SIP gateway cannot be configured on the same device. When CUBE license is active on the device, only CUBE will be functional.
- Only IPv4 is supported for transport and media (IPv4 transport for SIP registration, SIP signaling, and IPv4 packets encrypted over IPv4 transport).
- SIP signalling with peer devices behind NAT is not supported (ICE and STUN are not supported.
- SIP negotiation is supported only in global VRF.
- Remote-access VPN features like private address assignment, configuration mode exchange (CP payloads), routes exchange, are not supported.
- Routing protocols over the VPN-SIP session are not supported.
- Only Rivest-Shamir-Addleman (RSA) server self-signed certificates are supported.
- Pre-shared key lookup functionality using authentication, authorization, and accounting (AAA) is not supported.
- The IPSec idle timer is configured per IPSec profile using the ipsec-profile command. The idle time is the same for all VPN-SIP sessions that use a specific IPSec profile.
- Track objects that are used for IPSLA monitoring, have a maximum limit of 1000 objects in Cisco IOS software. When one track object is used to track one peer router, maximum number of VPN-SIP sessions that one IOS device can have is limited by the maximum number of track objects.
- Only one local SIP number is supported on Cisco IOS software.
- If there is a pre-existing statically configured policy, the programmatically configured QoS policy is not applied and session fails. Remove any statically configured QoS policy on the SVTI interface.
- On all Cisco ISR 1100 series routers, the supported scale of VPN-SIP feature is 300 sessions.
- Cisco does not support the interoperability with VPN-SIP implementation of other vendors.
- For the class policies included in the `policy-map` attached to the VPN-SIP tunnel, only Priority Queueing and Class-Based Weighted Fair Queueing (CBWFQ) are supported.
- For CBWFQ configurations, only the `bandwidth percent` *`percent`* command is supported. The `bandwidth` *`bandwidth`* command is not supported as the bandwidth of the VPN-SIP session varies depending on the negotiation with the peer router.

How to Configure VPN-SIP

Configuring VPN-SIP

The following steps describe the process of configuring VPN-SIP:

1. Configure the tunnel authentication using third party certificates, self-signed certificates, or pre-shared keys.

a. Tunnel Authentication using Certificates

Configure a trustpoint to obtain a certificate from a certification authority (CA) server that is located in the customer's network. This is required for tunnel authentication. Use the following configuration:

```
peer1(config)# crypto pki trustpoint CA
 enrollment url http://10.45.18.132/
 serial-number none
 subject-name CN=peer2
 revocation-check crl
 rsakeypair peer2

peer2(config)# crypto pki authenticate CA
Certificate has the following attributes:
       Fingerprint MD5: F38A9B4C 2D80490C F8E7581B BABE7CBD
      Fingerprint SHA1: 4907CC36 B1957258 5DFE23B2 649E7DDA 99BDB7C3
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.

peer2(config)#crypto pki enroll CA
%
% Start certificate enrollment ..
% Create a challenge password. You will need to verbally provide this
   password to the CA Administrator in order to revoke your certificate.
   For security reasons your password will not be saved in the configuration.
   Please make a note of it.
Password:
Re-enter password:
% The subject name in the certificate will include: CN=peer2
% The subject name in the certificate will include: peer2
% Include an IP address in the subject name? [no]:
Request certificate from CA? [yes/no]: yes
% Certificate request sent to Certificate Authority
% The 'show crypto pki certificate verbose CA' command will show the fingerprint.
Certificate map for Trustpoint
crypto pki certificate map data 1
issuer-name co cn = orange
```

b. Tunnel authentication using self-signed certificate

Configure a PKI trust point to generate a self-signed certificate on the device, when authenticating using a self-signed certificate. Use the following configuration:

```
peer4(config)#crypto pki trustpoint Self
    enrollment selfsigned
    revocation-check none
    rsakeypair myRSA
    exit
crypto pki enroll Self

Do you want to continue generating a new Self Signed Certificate? [yes/no]: yes
% Include the router serial number in the subject name? [yes/no]: yes
% Include an IP address in the subject name? [no]: no
Generate Self Signed Router Certificate? [yes/no]: yes

Router Self Signed Certificate successfully created
```

c. Configure tunnel authentication using a pre-shared key

```
crypto ikev2 keyring keys
peer peer1
identity key-id 1234
pre-shared-key key123
```

2. • Configure IKEv2 Profile for Certificate

```
crypto ikev2 profile IPROF
match certificate data
identity local key-id 5678
authentication remote rsa-sig
authentication local rsa-sig
keyring local keys
pki trustpoint self
nat force-encap
```

• Configure an IKEv2 Profile for pre-shared keys

```
crypto ikev2 profile IPROF
match identity remote any
identity local key-id 5678
authentication remote pre-share
authentication local pre-share
keyring local keys
nat force-encap
```

Note To complete the IKEv2 SA configuration, the **nat force-encap** command must be configured on both peers. Since, UDP encapsulation is negotiated in SDP, IKEv2 must start and continue on port 4500.

3. Configure an IPsec profile

```
crypto ipsec profile IPROF
set security-association idle-time 2000
```

4. Configure a LAN side interface

```
interface Vlan101
         ip address 192.0.2.3 255.255.255.0
         no shutdown
!
     interface GigabitEthernet2
         switchport access vlan 101
         no ip address
```

5. Configure a loopback interface

The loopback interface is used as the source interface for the secondary VPN tunnel.

```
interface loopback 1
    ip address 192.0.2.1 255.0.0.0
    ip nat inside
```

6. Configure a secondary interface.

Note Make sure the secondary interface is configured to receive the IP address, SIP server address, and vendor specific information via DHCP.

```
interface GigabitEthernet8
     ip dhcp client request sip-server-address
     ip dhcp client request vendor-identifying-specific
     ip address dhcp
    ip nat outside
```

7. Configure the tunnel interface

```
interface Tunnel1
     ip address 192.0.2.1 255.255.255.255
     load-interval 30
     tunnel source Loopback1
     tunnel mode ipsec ipv4
     tunnel destination dynamic
     tunnel protection ipsec profile IPROF ikev2-profile IPROF
     vpn-sip local-number 5678 remote-number 1234 bandwidth 1000
```

Use the **vpn-sip local-number** *local-number* **remote-number** *remote-number* **bandwidth** *bw-number* command to configure the sVTI interface for VPN-SIP. Bandwidth is the maximum data transmission rate that must be negotiated with this peer and the negotiated value is set on the tunnel interface. Allowed values are 64, 128, 256, 512, and 1000 kbps.

Once an SVTI is configured for VPN-SIP, changes cannot be made to tunnel mode, tunnel destination, tunnel source, and tunnel protection. To change the mode, source, destination, or tunnel protection you must remove the VPN-SIP configuration from the SVTI interface.

8. Add static routes to destination networks

Add a secondary route with a higher metric.

```
ip route 192.0.2.168 255.255.255.0 Tunnel0 track 1
ip route 192.0.2.168 255.255.255.0 Tunnel1 254
```

9. Configure IP SLA

```
ip sla 1
         icmp-echo 192.0.2.11
         threshold 500
         timeout 500
         frequency 2
        ip sla schedule 1 life forever start-time now
```

10. Configure route tracking

```
track 1 ip sla 1 reachability
```

11. Enable VPN-SIP

```
vpn-sip enable
vpn-sip local-number 5678 address ipv4 GigabitEthernet8
vpn-sip tunnel source Loopback1
vpn-sip logging
```

To configure VPN-SIP, you must configure local SIP number and local address. The **vpn-sip local-number** *SIP-number* **address ipv4** *WAN-interface-name* command configures the local SIP number that is used for SIP call and the associated IPv4 address.

Note Only IPv4 addresses can be configured. Crypto module does not support dual stack.

- Backup WAN interface address may change based on DHCP assignment.

When the primary WAN interface is functional, the destination of the VPN-SIP tunnel is set to the backup WAN interface, so that the tunnel interface is active. Destination is set to IP address of the peer that is learnt from SDP of SIP negotiation when traffic is routed to the tunnel interface. When primary WAN interface fails and the back routes are activated, packets are routed to the sVTI through backup.

Note We recommend that you use an unused non-routable address as the address of the loopback interface and do not configure this loopback interface for any other purpose. Once a loopback interface is configured, VPN-SIP listens to any updates to the interface and blocks them. The **vpn-sip logging** command enables the system logging of VPN-SIP module for events, such as session up, down, or failure.

Verifying VPN-SIP on a Local Router

Verifying Registration Status

```
Peer1# show vpn-sip registration-status
 SIP registration of local number 0388881001 : registered 10.6.6.50
```

Verifying SIP Registrar

```
Peer1#show vpn-sip sip registrar

Line          destination      expires(sec)  contact   transport     call-id
==========================================================================================
0388881001    example.com         2359        10.6.6.50  UDP
3176F988-9EAA11E7-8002AFA0-8EF41435
```

Verifying VPN-SIP Status

```
Peer1#show vpn-sip session detail
VPN-SIP session current status

Interface: Tunnel1
   Session status: SESSION_UP (I)
   Uptime        : 00:00:42
   Remote number : 0388881001 =====> This is the Remote Router's SIP number
   Local number  : 0388882001 =====> Local router's SIP number
   Remote address:port: 10.6.6.49:50002
   Local address:port : 10.6.6.50:50001
   Crypto conn handle: 0x8000017D
   SIP Handle        : 0x800000C7
   SIP callID        : 1554
   Configured/Negotiated bandwidth: 64/64 kbps
```

Verifying Crypto Session

```
Peer1# show crypto session detail
Crypto session current status
Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
X - IKE Extended Authentication, F - IKE Fragmentation
R - IKE Auto Reconnect, U - IKE Dynamic Route Update
S - SIP Vpn-sip

Interface: Tunnel1
Profile: IPROF
Uptime: 00:03:53
Session status: UP-ACTIVE
Peer: 10.6.6.49 port 4500 fvrf: (none) ivrf: (none)
      Phase1_id: 10.6.6.49
      Desc: (none)
```

```
  Session ID: 43
  IKEv2 SA: local 10.11.1.1/4500 remote 10.6.6.49/50002 Active
          Capabilities:S connid:1 lifetime:23:56:07 ====> Capabilities:S indicates this is
 a  SIP VPN_SIP Session
  IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
        Active SAs: 2, origin: crypto map
        Inbound:  #pkts dec'ed 6 drop 0 life (KB/Sec) 4222536/3366
        Outbound: #pkts enc'ed 4 drop 0 life (KB/Sec) 4222537/3366
```

Verifying IP NAT Translations

```
Peer1#sh ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
udp 2.2.2.2:4500       10.6.6.50:50001    10.6.6.49:50002    10.6.6.49:50002
```

Verifying DHCP SIP Configuration

```
Peer9#show vpn-sip sip dhcp
SIP DHCP Info

SIP-DHCP interface:  GigabitEthernet8

SIP server address:
Domain name:         dns:example.com
```

Verifying VPN-SIP on a Remote Router

Verifying VPN-SIP Registration Status on a Remote Router

```
Peer2# show vpn-sip registration-status
 SIP registration of local number 0388882001 : registered 10.6.6.49
```

Verifying VPN-SIP Registrar on a Remote Router

```
Peer2# show vpn-sip sip registrar
Line          destination      expires(sec)  contact    transport    call-id
==========================================================================================
0388882001    example.com    2478         10.6.6.49   UDP
E6F23809-9EAB11E7-80029279-40B97F59
```

Verifying VPN-SIP Session Details on a Remote Router

```
Peer2# show vpn-sip session detail
VPN-SIP session current status
Interface: Tunnel1
   Session status: SESSION_UP (R)
   Uptime        : 00:00:21
   Remote number : 0388882001 =====> This is the Peer1 Router's SIP number
   Local number  : 0388881001 =====> Local router's SIP number
   Remote address:port: 10.6.6.50:50001
   Local address:port : 10.6.6.49:50002
   Crypto conn handle: 0x8000017E
   SIP Handle        : 0x800000BE
   SIP callID        : 1556
   Configured/Negotiated bandwidth: 1000/64 kbps
```

Verifying Crypto Session Details on a Remote Router

```
Peer2 #show crypto session detail
Crypto session current status
```

```
Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
X - IKE Extended Authentication, F - IKE Fragmentation
R - IKE Auto Reconnect, U - IKE Dynamic Route Update
S - SIP VPN-SIP

Interface: Tunnel1
Profile: IPROF
Uptime: 00:02:32
Session status: UP-ACTIVE
Peer: 10.6.6.50 port 50001 fvrf: (none) ivrf: (none)
      Phase1_id: 10.6.6.50
      Desc: (none)
  Session ID: 147
  IKEv2 SA: local 10.17.1.1/4500 remote 10.6.6.50/50001 Active
          Capabilities:S connid:1 lifetime:23:57:28 ====> Capabilities:S indicates this is
 a  SIP VPN-SIP Session
  IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
        Active SAs: 2, origin: crypto map
        Inbound:  #pkts dec'ed 4 drop 0 life (KB/Sec) 4293728/3448
        Outbound: #pkts enc'ed 6 drop 0 life (KB/Sec) 4293728/3448
```

Verifying IP NAT Translations on a Remote Router

```
Peer2#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
udp 3.3.3.3:4500       10.6.6.49:50002    10.6.6.50:50001    10.6.6.50:50001
```

Configuring QoS for VPN-SIP

Optionally, you can apply a quality of service (QoS) policy to the VPN-SIP. A QoS policy provides secure, predictable, measurable, and sometimes guaranteed services to certain types of traffic.

1. Configure the appropriate policy map.

```
Device(config)#class-map match-all UDP
 match protocol ip
!
policy-map CBWFQ
 class UDP
  bandwidth percent 60
  queue-limit 12 packets
```

2. Attach the policy-map to the VPN-SIP:

```
Device(config)#interface Tunnel1
.
.
.
vpn-sip local-number 5678 remote-number 1234 bandwidth 1000 service-policy CBWFQ
```

Note When the VPN-SIP session is successfully negotiated and comes up, an implicit service policy is automatically attached to the tunnel interface. If you run the `show running-config` command for this interface, the implicit service policy is not displayed. Any `policy-map` that you create on the device becomes a child policy of this implicit service policy.

Verifying QoS for VPN-SIP

Verifying the Application of the Policy Map

```
Peer1#sh policy-map int tun1
 Tunnel1

  Service-policy output: VPN-SIP-Tunnel1-Bandwidth

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      QoS Set
        dscp cs4
          Packets marked 0
      shape (average) cir 1000000, bc 4000, be 4000
      target shape rate 1000000

      Service-policy : CBWFQ

        Class-map: UDP (match-all)
          0 packets, 0 bytes
          5 minute offered rate 0000 bps, drop rate 0000 bps
          Match: protocol ip
          Queueing
          queue limit 12 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
          bandwidth 60% (600 kbps)


        Class-map: class-default (match-any)
          0 packets, 0 bytes
          5 minute offered rate 0000 bps, drop rate 0000 bps
          Match: any

          queue limit 64 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
Peer1#sh vpn-sip session detail
VPN-SIP session current status

Interface: Tunnel1
   Session status: SESSION_UP (R)
   Uptime        : 00:00:15
   Remote number : 5678
   Local number  : 1234
   Remote address:port: 6.6.6.40:51878
   Local address:port : 6.6.6.89:50010
   Crypto conn handle: 0x40000017
   SIP Handle        : 0x4000000B
   SIP callID        : 2288
   Configured/Negotiated bandwidth: 1000/1000 kbps
   Applied service policy: CBWFQ
```

Verifying the Flow of Traffic

After sending UDP traffic in the direction of the policy, verify the flow of traffic as follows:

```
Peer1#sh policy-map int tun1
 Tunnel1

  Service-policy output: VPN-SIP-Tunnel1-Bandwidth

    Class-map: class-default (match-any)
      105782 packets, 4865972 bytes
      5 minute offered rate 130000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/98707/0
      (pkts output/bytes output) 7068/890568
      QoS Set
        dscp cs4
          Packets marked 105782
      shape (average) cir 1000000, bc 4000, be 4000
      target shape rate 1000000

      Service-policy : CBWFQ

        Class-map: UDP (match-all)
          105775 packets, 4865650 bytes
          5 minute offered rate 130000 bps, drop rate 331000 bps
          Match: protocol ip
          Queueing
          queue limit 12 packets
          (queue depth/total drops/no-buffer drops) 11/98707/0
          (pkts output/bytes output) 7068/890568
          bandwidth 60% (600 kbps)


        Class-map: class-default (match-any)
          0 packets, 0 bytes
          5 minute offered rate 0000 bps, drop rate 0000 bps
          Match: any

          queue limit 64 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 0/0
```

Configuration Examples for VPN-SIP

Using self-signed certificates for authentication

The following is sample configuration to configure VPN-SIP using self-signed certificates for authentication. There is no distinction between initiator and responder role in VPN-SIP. The configuration on a peer node will be identical with local SIP numbers changed.

```
// Self-signed certificate
crypto pki trustpoint selfCert
  rsakeypair myRSA
  enrollment selfsigned
  revocation-check none
!
crypto ikev2 profile vpn-sip-profile
```

```
 match identity remote any
 authentication local rsa-sig
 authentication remote rsa-sig
 pki trustpoint selfCert // Use same self-signed trustpoint for sign and verify
 nat force-encap
!
crypto ipsec profile vpn-sip-ipsec
 set security-association idle-time 120
!
vpn-sip enable
vpn-sip local-number 0388883001 address ipv4 GigabitEthernet1
vpn-sip tunnel source Loopback11
vpn-sip logging
!
// one tunnel per peer – configuration is for peer with a SIP-number of 0388884001
int tunnel0
 ip unnumbered loopback 0
 tunnel source loopback11
 tunnel mode ipsec ipv4
 tunnel destination dynamic
 tunnel protection ipsec profile vpn-sip-ipsec ikev2-profile vpn-sip-profile
 vpn-sip local-number 0388883001 remote-number 0388884001 bandwidth 1000
!
// ip unnumbered of tunnel interfaces
int loopback 0
  ip address 10.21.1.1 255.255.255.255
!
int loopback11
ip address 10.9.9.9 255.255.255.255
ip nat inside
!
// one tunnel per peer – this is for peer with SIP-number 0388885001
int tunnel1
 ip unnumbered loopback 0
 tunnel source loopback11
 tunnel mode ipsec ipv4
 tunnel destination dynamic
 tunnel protection ipsec profile vpn-sip-ipsec ikev2-profile iprof
 vpn-sip sip-local 0388883001 sip-remote 0388885001 bandwidth 1000
!
interface GigabitEthernet8
 ip dhcp client request sip-server-address
 ip dhcp client request vendor-identifying-specific
 ip address dhcp
ip nat outside


// backup routes configured with higher AD so that these routes will be activated only when
 primary path goes down. AD need to be chosen to be greater than that of primary route.
ip route 10.0.0.0 255.0.0.0 tunnel 0 250
ip route 10.1.0.0 255.0.0.0 tunnel 0 250
ip route 10.2.0.0 255.0.0.0 tunnel 0 250
ip route 10.3.0.0 255.0.0.0 tunnel 0 250
```

Troubleshooting for VPN-SIP

Viewing Tunnel Interface in Show Output

Symptom

Show VPN-SIP session doesn't show any information about the tunnel interface. In the following example, information about the tunnel interface, tunnel1 is not shown:

```
Peer5-F#show vpn-sip session
VPN-SIP session current status

Interface: Tunnel2
   Session status: READY_TO_CONNECT
   Remote number : 0334563333
   Local number  : 0623458888
   Remote address:port: 10.10.0.0:0
   Local address:port : 192.0.2.22:0

Interface: Tunnel3
   Session status: READY_TO_CONNECT
   Remote number : 0323452222
   Local number  : 0623458888
   Remote address:port: 10.10.0.0:0
   Local address:port : 192.0.2.22:0

Interface: Tunnel4
   Session status: READY_TO_CONNECT
   Remote number : 0612349999
   Local number  : 0623458888
   Remote address:port: 10.10.0.0:0
   Local address:port : 192.0.2.22:0

Interface: Tunnel6
   Session status: READY_TO_CONNECT
   Remote number : 0634567777
   Local number  : 0623458888
   Remote address:port: 10.10.0.0:0
   Local address:port : 172.30.18.22:0
```

Possible Cause

VPN-SIP is not configured on the tunnel interface

```
Peer5-F#sh run int tun1
Building configuration...

Current configuration : 201 bytes
!
interface Tunnel1
 ip address 10.5.5.5 255.255.255.0
 tunnel source Loopback11
 tunnel mode ipsec ipv4
 tunnel destination dynamic
 tunnel protection ipsec profile test-prof ikev2-profile test
end
```

Recommended Action

Configure VPN-SIP on the tunnel interface.

```
:

Peer5-F#show running interface tunnel 1
Building configuration...

Current configuration : 278 bytes
!
interface Tunnel1
 ip address 10.5.5.5 255.255.255.255
 tunnel source Loopback11
```

```
 tunnel mode ipsec ipv4
 tunnel destination dynamic
 tunnel protection ipsec profile test-prof ikev2-profile test
 vpn-sip local-number 0623458888 remote-number 0312341111 bandwidth 1000
end
```

Following is the running output for the above scenario:

```
Peer5-F#show vpn-sip session detail
VPN-SIP session current status

Interface: Tunnel1
   Session status: READY_TO_CONNECT
   Remote number : 0312341111
   Local number  : 0623458888
   Remote address:port: 10.0.0.0:0
   Local address:port : 172.30.18.22:0

   Crypto conn handle: 0x8000002C
   SIP Handle        : 0x0
   SIP callID        : --
   Configured/Negotiated bandwidth: 1000/0 kbps

Interface: Tunnel2
   Session status: READY_TO_CONNECT
   Remote number : 0334563333
   Local number  : 0623458888
   Remote address:port: 10.0.0.0:0
   Local address:port : 172.30.18.22:0
   Crypto conn handle: 0x80000012
   SIP Handle        : 0x0
   SIP callID        : --
   Configured/Negotiated bandwidth: 512/0 kbps

Interface: Tunnel3
   Session status: READY_TO_CONNECT
   Remote number : 0323452222
   Local number  : 0623458888
   Remote address:port: 10.0.0.0:0
   Local address:port : 172.30.18.22:0
   Crypto conn handle: 0x80000031
   SIP Handle        : 0x0
   SIP callID        : --
   Configured/Negotiated bandwidth: 512/0 kbps

Interface: Tunnel4
   Session status: READY_TO_CONNECT
   Remote number : 0612349999
   Local number  : 0623458888
   Remote address:port: 10.0.0.0:0
   Local address:port : 172.30.18.22:0
   Crypto conn handle: 0x8000002F
   SIP Handle        : 0x0
   SIP callID        : --
   Configured/Negotiated bandwidth: 1000/0 kbps

Interface: Tunnel6
   Session status: READY_TO_CONNECT
   Remote number : 0634567777
   Local number  : 0623458888
   Remote address:port: 10.0.0.0:0
   Local address:port : 172.30.18.22:0
   Crypto conn handle: 0x80000026
   SIP Handle        : 0x0
```

```
   SIP callID        : --
   Configured/Negotiated bandwidth: 1000/0 kbps
```

Troubleshooting SIP Registration Status

Symptom

SIP registration status is Not Registered

```
Peer5#show vpn-sip sip registrar
Line          destination      expires(sec)  contact
transport     call-id
============================================================

Peer5-F#show vpn-sip registration-status

 SIP registration of local number 0623458888 : not registered
```

Possible Cause

IP address is not configured on the WAN interface.

```
Peer5#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     unassigned      YES unset  down                  down
GigabitEthernet0/1     unassigned      YES unset  up                    up
GigabitEthernet0/2     unassigned      YES unset  down                  down
GigabitEthernet0/3     unassigned      YES unset  down                  down
GigabitEthernet0/4     unassigned      YES unset  up                    up
GigabitEthernet0/5     10.5.5.5        YES manual up                    up
Vlan1                  10.45.1.5       YES NVRAM  up                    up
NVI0                   10.1.1.1        YES unset  up                    up
Loopback1              10.1.1.1        YES NVRAM  up                    up
Loopback5              10.5.5.5        YES NVRAM  administratively down down
Loopback11             10.11.11.11     YES NVRAM  up                    up
Tunnel1                10.5.5.5        YES NVRAM  up                    down
Tunnel2                10.2.2.2        YES NVRAM  up                    down
Tunnel3                10.3.3.3        YES NVRAM  up                    down
Tunnel4                10.4.4.4        YES NVRAM  up                    down
Tunnel6                10.8.8.8        YES NVRAM  up                    down

Peer5-F#show run interface gigabitEthernet 0/4
Building configuration...

Current configuration : 213 bytes
!
interface GigabitEthernet0/4
 ip dhcp client request sip-server-address
 ip dhcp client request vendor-identifying-specific
 no ip address          ====> no IP address
 ip nat outside
 ip virtual-reassembly in
 duplex auto
 speed auto
end
```

Recommended Action

Use the **ip address dhcp** command to configure the interface IP address.

```
Peer5-F#show running-config interface gigabitEthernet 0/4
Building configuration...

Current configuration : 215 bytes
```

```
!
interface GigabitEthernet0/4
 ip dhcp client request sip-server-address
 ip dhcp client request vendor-identifying-specific
 ip address dhcp         ====> configure IP address DHCP
 ip nat outside
 ip virtual-reassembly in
 duplex auto
 speed auto
end


Peer5-F#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     unassigned      YES unset  down                  down
GigabitEthernet0/1     unassigned      YES unset  up                    up
GigabitEthernet0/2     unassigned      YES unset  down                  down
GigabitEthernet0/3     unassigned      YES unset  down                  down
GigabitEthernet0/4     172.30.18.22    YES DHCP   up                    up
GigabitEthernet0/5     10.5.5.5        YES manual up                    up
Vlan1                  10.45.1.5       YES NVRAM  up                    up
NVI0                   10.1.1.1        YES unset  up                    up
Loopback1              10.1.1.1        YES NVRAM  up                    up
Loopback5              10.5.5.5        YES NVRAM  administratively down down
Loopback11             10.11.11.11     YES NVRAM  up                    up
Tunnel1                10.6.5.5        YES NVRAM  up                    down
Tunnel2                10.2.2.2        YES NVRAM  up                    down
Tunnel3                10.3.3.3        YES NVRAM  up                    down
Tunnel4                10.4.4.4        YES NVRAM  up                    down
Tunnel6                10.8.8.8        YES NVRAM  up                    down

Peer5-F#show vpn-sip sip registrar
Line          destination      expires(sec)  contact
transport     call-id
============================================================
0623458888    example.com   2863          172.30.18.22
UDP           1E83ECF0-AF0611E7-802B8FCF-594EB9E7@10.50.18.22

Peer5-F#show vpn-sip registration-status

 SIP registration of local number 0623458888 : registered 172.30.18.22
```

Session stuck in Negotiating IKE state

Symptom

VPN-SIP session stuck in Negotiating IKE state.

```
Peer5#show vpn-sip session remote-number 0612349999 detail
VPN-SIP session current status

Interface: Tunnel4
   Session status: NEGOTIATING_IKE (R)
   Uptime        : 00:00:58
   Remote number : 0612349999
   Local number  : 0623458888
   Remote address:port: 72.30.168.3:24825
   Local address:port : 72.30.168.22:50012
   Crypto conn handle: 0x8000002E
   SIP Handle        : 0x8000000C
   SIP callID        : 16
   Configured/Negotiated bandwidth: 1000/1000 kbps
```

Possible Cause

Bad configuration related to IKEv2.

In the following example the Key ID that is configured in the keyring does not match the SIP numberof the remote peer.

```
Peer5-F#show running-config interface tunnel 4
Building configuration...

Current configuration : 276 bytes
!
interface Tunnel4
 ip address 10.4.4.4 255.255.255.0
 tunnel source Loopback11
 tunnel mode ipsec ipv4
 tunnel destination dynamic
 tunnel protection ipsec profile test-prof ikev2-profile test
 VPN-SIP local-number 0623458888 remote-number 0612349999 bandwidth 1000  ====> Remote
number mentioned here doesn’t match the remote number in the keyring
end

IKEv2 Keyring configs:
!
crypto ikev2 keyring keys
 peer peer1
  identity key-id 0312341111
  pre-shared-key psk1
 !
 peer abc
  identity key-id 0345674444
  pre-shared-key psk1
 !
 peer peer2
  identity key-id 0334563333
  pre-shared-key psk10337101690
 !
 peer peer6
  identity key-id 0634567777
  pre-shared-key cisco123
 !
 peer peer3
  identity key-id 0323452222
  pre-shared-key cisco123
 !
 peer peer4
  identity key-id 0645676666
  pre-shared-key psk1
 !
 peer NONID
  identity fqdn example.com
  pre-shared-key psk1
 !
!
!
crypto ikev2 profile test
 match identity remote any
 identity local key-id 0623458888
 authentication remote pre-share
 authentication local pre-share
 keyring local keys
 dpd 10 6 periodic
 nat force-encap
```

Recommended Action

Correct the keyring configurations.

```
rypto ikev2 keyring keys
 peer peer1
  identity key-id 0312341111
  pre-shared-key psk1
 !
 peer abc
  identity key-id 0345674444
  pre-shared-key psk1
 !
 peer peer2
  identity key-id 0334563333
  pre-shared-key psk1
 !
 peer peer6
  identity key-id 0634567777
  pre-shared-key psk1
 !
 peer peer3
  identity key-id 0323452222
  pre-shared-key psk1
 !
 peer peer4
  identity key-id 0612349999
  pre-shared-key psk1
 !
 peer NONID
  identity fqdn example.com
  pre-shared-key psk1
 !
!
!
crypto ikev2 profile test
 match identity remote any
 identity local key-id 0623458888
 authentication remote pre-share
 authentication local pre-share
 keyring local keys
 dpd 10 6 periodic
 nat force-encap
!

Peer5-F#show vpn-sip session remote-number  0612349999 detail
VPN-SIP session current status

Interface: Tunnel4
   Session status: SESSION_UP (R)
   Uptime        : 00:02:04
   Remote number : 0612349999
   Local number  : 0623458888
   Remote address:port: 198.51.100.3:24845
   Local address:port : 198.51.100.22:50020
   Crypto conn handle: 0x8000004E
   SIP Handle        : 0x80000014
   SIP callID        : 24
   Configured/Negotiated bandwidth: 1000/1000 kbps
```

Troubleshooting Session Initiation

Symptom

Session does not initiate and gets stuck in Negotiating IKE state

Possible Cause

Fagmentation of IKE packets when a large PKI certificate is included in the IKE authentication message.

Recommended Action

Configure IKEv2 fragmentation on the routers.

Debug Commands

The folllwing debug commands are available to debug VPN-SIP configuration:

Table 43: debug commands

Command Name	Description
debug vpn-sip event	Prints debug messages for SVTI registration with VPN-SIP, SIP registration, call setup, and so on.
debug vpn-sip errors	Prints error messages only when an error occurs during initialization, registration, call setup, and so on.
debug vpn-sip sip all	Enables all SIP debugging traces.
debug vpn-sip sip calls	Enables SIP SPI calls debugging trace.
debug vpn-sip sip dhcp	Enables SIP-DHCP debugging trace
debug vpn-sip sip error	Enables SIP error debugging trace
debug vpn-sip sip events	Enables SIP events debugging trace.
debug vpn-sip sip feature	Enables feature level debugging.
debug vpn-sip sip function	Enables SIP function debugging trace.
debug vpn-sip sip info	Enables SIP information debugging trace.
debug vpn-sip sip level	Enables information level debugging.
debug vpn-sip sip media	Enables SIP media debugging trace.
debug vpn-sip sip messages	Enables SIP SPI messages debugging trace
debug vpn-sip sip non-call	Enables Non-Call-Context trace (OPTIONS, SUBSCRIBE, and so on)
debug vpn-sip sip preauth	Enable SIP preauth debugging trace.
debug vpn-sip sip states	Enable SIP SPI states debugging trace.
debug vpn-sip sip translate	Enables SIP translation debugging trace.
debug vpn-sip sip transport	Enables SIP transport debugging traces.
debug vpn-sip sip verbose	Enables verbose mode.

Additional References for VPN-SIP

Standards and RFCs

Standard/RFC	Title
RFC 6193 (with Restrictions)	Media Description for the Internet Key Exchange Protocol (IKE) in the Session Description Protocol (SDP)

Feature Information for VPN-SIP

Table 44: Feature Information for VPN-SIP

Feature Name	Releases	Feature Information
Session Initiation Protocol Triggered VPN		VPN-SIP is a service offered by service providers where a VPN is setup for on-demand media or application sharing between peers, using Session Initiation Protocol (SIP). The following commands were introduced: **nat force-encap, show vpn-sip session, show vpn-sip sip, show vpn-sip registration-status, vpn-sip local-number, vpn-sip logging, vpn-sip tunnel source**.

CHAPTER 28

Configuring Voice Functionality

This chapter provides information about configuring voice functionality on the Cisco 4000 Series Integrated Services Routers (ISRs).

This chapter includes these sections:

Call Waiting

With the Call Waiting feature, you can receive a second call while you are on the phone attending to another call. When you receive a second call, you hear a call-waiting tone (a tone with a 300 ms duration). Caller ID appears on phones that support caller ID. You can use hookflash to answer a waiting call and place the previously active call on hold. By using hookflash, you can toggle between the active and a call that is on hold. If the Call Waiting feature is disabled, and you hang up the current call, the second call will hear a busy tone. For more information on Call Waiting, see http://www.cisco.com/c/en/us/td/docs/ios/voice/sip/configuration/guide/15_0/sip_15_0_book/sip_cg-hookflash.html#wp999028

Call Transfers

Call transfers are when active calls are put on hold while a second call is established between two users. After you establish the second call and terminate the active call, the call on hold will hear a ringback. The Call Transfer feature supports all three types of call transfers—blind, semi-attended, and attended. For more information on Call Transfers, see the http://www.cisco.com/c/en/us/td/docs/ios/voice/sip/configuration/guide/15_0/sip_15_0_book/sip_cg-hookflash.html#wp999084

E1 R2 Signaling Configuration

To configure the E1 R2, perform these steps:

Before you begin

Before you attempt this configuration, ensure that you meet these prerequisites:

- R2 signaling applies only to E1 controllers.
- In order to run R2 signaling on Cisco 4000 Series ISRs, this hardware is required:
- NIM-MFT-1T1/E1 or NIM-2MFT-T1/E1 or NIM-4MFT-T1/E1or NIM-8MFT-T1/E1 or NIM-1CE1T1-PRI or NIM-2CE1T1-PRI or NIM-8CE1T1-PRI
- Define the command ds0-group on the E1 controllers of Cisco 4000 Series ISRs.
- Cisco IOS XE software release 15.5 (2)

SUMMARY STEPS

1. Set up the controller E1 that connects to the private automatic branch exchange (PBX) or switch.
2. For E1 framing, choose either **CRC** or **non-CRC**
3. For E1 linecoding, choose either **HDB3** or **AMI**.
4. For the E1 clock source, choose either internal or line. Note that different PBXs have different requirements on the clock source.
5. Configure line signaling.
6. Configure interregister signaling.
7. Customize the configuration with the cas-custom command.

DETAILED STEPS

Step 1 Set up the controller E1 that connects to the private automatic branch exchange (PBX) or switch.

Ensure that the framing and linecoding of the E1 are properly set.

Step 2 For E1 framing, choose either **CRC** or **non-CRC**

Step 3 For E1 linecoding, choose either **HDB3** or **AMI**.

Step 4 For the E1 clock source, choose either internal or line. Note that different PBXs have different requirements on the clock source.

Step 5 Configure line signaling.

```
(config)# controller E1 0/2/0


 (config-controller)#ds0-group 1 timeslots 1 type ?
...
r2-analog        R2 ITU Q411
r2-digital       R2 ITU Q421
r2-pulse         R2 ITU Supplement 7
...
```

Step 6 Configure interregister signaling.

```
(config)# controller E1 0/2/0

 eefje(config)# controller E1 0/2/0
 eefje(config-controller)#ds0-group 1 timeslots 1 type r2-digital ?
dtmf                DTMF tone signaling
r2-compelled        R2 Compelled Register Signaling
```

```
r2-non-compelled   R2 Non Compelled Register Signaling
r2-semi-compelled  R2 Semi Compelled Register Signaling

...
```

The Cisco implementation of R2 signaling has Dialed Number Identification Service (DNIS) support enabled by default. If you enable the Automatic Number Identification (ANI) option, the collection of DNIS information is still performed. Specification of the ANI option does not disable DNIS collection. DNIS is the number that is called and ANI is the number of the caller. For example, if you configure a router called A to call a router called B, then the DNIS number is assigned to router B and the ANI number is assigned to router A. ANI is similar to caller ID.

Step 7 Customize the configuration with the cas-custom command.

```
(config)# controller E1 0/2/0


(config-controller)#ds0-group 1 timeslots 1 type r2-digital r2-compelled ani
cas-custom 1
  country brazil
  metering
  answer-signal group-b 1

voice-port 0/2/0:1
!
dial-peer voice 200 pots
destination-pattern 43200
direct-inward-dial
port 0/2/0:1

dial-peer voice 3925 voip
destination-pattern 39...
session target ipv4:10.5.25.41
...
```

R2 Configurations

The configurations have been modified in order to show only the information that this document discusses.

Configured for R2 Digital Non-Compelled

```
hostname eefje
!
controller E1 0
 clock source line primary
 dso-group 1 timeslots 1-15 type r2-digital r2-non-compelled
 cas-custom 1

!--- For more information on these commands
!--- refer to
ds0-group
 and
cas-custom.

!
voice-port 0:1
 cptone BE

!--- The cptone command is country specific. For more
!--- information on this command,  refer to
```

```
cptone
.

!
dial-peer voice 123 pots
 destination-pattern 123
 direct-inward-dial
 port 0:1
 prefix 123
!
dial-peer voice 567 voip
 destination-pattern 567
 session target ipv4:10.0.0.2
```

Configured for R2 Digital Semi-Compelled

```
hostname eefje
!
controller E1 0
 clock source line primary
 ds0-group 1 timeslots 1-15 type r2-digital r2-semi-compelled
 cas-custom 1

!--- For more information on these commands
!---  refer to
ds0-group
 and
cas-custom
.

!
voice-port 0:1
 cptone BE

!--- The cptone command is country specific. For more
!--- information on this command,  refer to
cptone
.

dial-peer voice 123 pots
 destination-pattern 123
 direct-inward-dial
 port 0:1
 prefix 123
!
dial-peer voice 567 voip
 destination-pattern 567
 session target ipv4:10.0.0.2
```

Configured for R2 Digital Compelled ANI

```
hostname eefje
! controller E1 0 clock source line primary ds0-group
1 timeslots 1-15 type r2-digital r2-compelled ani cas-custom 1

!--- For more information on these commands
!---  refer to
ds0-group
 and
cas-custom
.

voice-port 0:1 cptone BE

!--- The cptone command is country specific. For more
!--- information on this command,  refer to
```

```
cptone
.

dial-peer voice 123 pots destination-pattern 123 direct-inward-dial port
0:1 prefix 123
!
dial-peer voice 567 voip destination-pattern 567 session
target ipv4:10.0.0.2
```

Sample Debug Command Output

This example shows the output for the **debug vpm sig** command.

```
(config-controller)#debug vpm sig
Syslog logging: enabled
(0 messages dropped, 9 messages rate-limited, 1 flushes, 0 overruns,
 xml disabled, filtering disabled)No Active Message Discriminator.
No Inactive Message Discriminator.
Console logging: disabled
Monitor logging: level debugging, 0 messages logged, xml disabled, filtering disabled
Buffer logging:  level debugging, 163274 messages logged, xml disabled,filtering disabled

Exception Logging: size (4096 bytes)    Count and timestamp logging messages: disabled
Persistent logging: disabledNo active filter modules.
Trap logging: level informational, 172 message lines logged
Logging Source-Interface:
VRF Name:Log Buffer (4096 bytes):0): DSX (E1 0/2/0:0): STATE: R2_IN_COLLECT_DNIS R2 Got
Event 1
*Jan 29 21:32:22.258:r2_reg_generate_digits(0/2/0:1(1)): Tx digit '1'
*Jan 29 21:32:22.369: htsp_digit_ready(0/2/0:1(1)): Rx digit='#'
*Jan 29 21:32:22.369: R2 Incoming Voice(0/0): DSX (E1 0/2/0:0):STATE: R2_IN_COLLECT_DNIS
R2 Got Event R2_TONE_OFF
*Jan 29 21:32:22.369: r2_reg_generate_digits(0/2/0:1(1)): Tx digit '#'
*Jan 29 21:32:22.569: htsp_dialing_done(0/2/0:1(1))
*Jan 29 21:32:25.258: R2 Incoming Voice(0/0): DSX (E1 0/2/0:0):STATE: R2_IN_COLLECT_DNIS
R2 Got Event R2_TONE_TIMER
*Jan 29 21:32:25.258: r2_reg_generate_digits(0/2/0:1(1)): Tx digit '3#'
*Jan 29 21:32:25.520: htsp_digit_ready_up(0/2/0:1(1)): Rx digit='1'
*Jan 29 21:32:25.520: R2 Incoming Voice(0/0): DSX (E1 0/2/0:0): STATE: R2_IN_CATEGORY R2
Got Event 1
*Jan 29 21:32:25.520: Enter r2_comp_category
*Jan 29 21:32:25.520: R2 Event : 1
*Jan 29 21:32:25.520:  ####### collect_call_enable = 0
*Jan 29 21:32:25.520: ######## Not Sending B7 ##################
*Jan 29 21:32:25.520: r2_reg_event_proc(0/2/0:1(1)) ADDR_INFO_COLLECTED (DNIS=39001,
ANI=39700)
*Jan 29 21:32:25.520: r2_reg_process_event: [0/2/0:1(1), R2_REG_COLLECTING,
E_R2_REG_ADDR_COLLECTED(89)]
*Jan 29 21:32:25.520: r2_reg_ic_addr_collected(0/2/0:1(1))htsp_switch_ind
*Jan 29 21:32:25.521: htsp_process_event: [0/2/0:1(1), R2_Q421_IC_WAIT_ANSWER,
E_HTSP_SETUP_ACK]
*Jan 29 21:32:25.521: r2_q421_ic_setup_ack(0/2/0:1(1)) E_HTSP_SETUP_ACK
*Jan 29 21:32:25.521: r2_reg_switch(0/2/0:1(1))
*Jan 29 21:32:25.521: r2_reg_process_event: [0/2/0:1(1), R2_REG_WAIT_FOR_SWITCH,
E_R2_REG_SWITCH(96)]
*Jan 29 21:32:25.521: r2_reg_ic_switched(0/2/0:1(1))
*Jan 29 21:32:25.522: htsp_process_event: [0/2/0:1(1), R2_Q421_IC_WAIT_ANSWER,
E_HTSP_PROCEEDING]
*Jan 29 21:32:25.530:htsp_call_bridged invoked
*Jan 29 21:32:25.530: r2_reg_event_proc(0/2/0:1(1)) ALERTING RECEIVED
*Jan 29 21:32:25.530: R2 Incoming Voice(0/0): DSX (E1 0/2/0:0): STATE: R2_IN_WAIT_REMOTE_ALERT
 R2 Got Event R2_ALERTING
*Jan 29 21:32:25.530:rx R2_ALERTING in r2_comp_wait_remote_alert
*Jan 29 21:32:25.530: r2_reg_generate_digits(0/2/0:1(1)): Tx digit '1'htsp_alert_notify
```

```
*Jan 29 21:32:25.531:r2_reg_event_proc(0/2/0:1(1)) ALERTING RECEIVED
*Jan 29 21:32:25.531: R2 Incoming Voice(0/0): DSX (E1 0/2/0:0): STATE: R2_IN_COMPLETE R2
Got Event R2_ALERTING
*Jan 29 21:32:25.540: htsp_dsp_message: RESP_SIG_STATUS: state=0x0 timestamp=0
systime=80352360
*Jan 29 21:32:25.540:htsp_process_event: [0/2/0:1(1), R2_Q421_IC_WAIT_ANSWER, E_DSP_SIG_0000]
*Jan 29 21:32:25.651: htsp_dialing_done(0/2/0:1(1))
*Jan 29 21:32:25.751: htsp_digit_ready(0/2/0:1(1)): Rx digit='#'
*Jan 29 21:32:25.751: R2 Incoming Voice(0/0): DSX (E1 0/2/0:0): STATE: R2_IN_COMPLETE R2
Got Event R2_TONE_OFF
*Jan 29 21:32:25.751: r2_reg_generate_digits(0/2/0:1(1)): Tx digit '#'
*Jan 29 21:32:25.961: htsp_dialing_done(0/2/0:1(1))
*Jan 29 21:32:26.752: R2 Incoming Voice(0/0): DSX (E1 0/2/0:0): STATE: R2_IN_WAIT_GUARD R2
 Got Event R2_TONE_TIMER
*Jan 29 21:32:26.752: R2_IN_CONNECT: call end dial
*Jan 29 21:32:26.752: r2_reg_end_dial(0/2/0:1(1))htsp_call_service_msghtsp_call_service_msg
 not EFXS (11)htsp_call_service_msghtsp_call_service_msg not EFXS (11)
*Jan 29 21:32:26.754: htsp_process_event: [0/2/0:1(1), R2_Q421_IC_WAIT_ANSWER,
E_HTSP_VOICE_CUT_THROUGH]
*Jan 29 21:32:26.754: htsp_process_event: [0/2/0:1(1), R2_Q421_IC_WAIT_ANSWER,
E_HTSP_VOICE_CUT_THROUGH]
*Jan 29 21:32:26.754: htsp_process_event: [0/2/0:1(1), R2_Q421_IC_WAIT_ANSWER,
E_HTSP_VOICE_CUT_THROUGH]
*Jan 29 21:32:51.909: htsp_process_event: [0/2/0:1(1), R2_Q421_IC_WAIT_ANSWER, E_HTSP_CONNECT]
*Jan 29 21:32:51.909: r2_q421_ic_answer(0/2/0:1(1)) E_HTSP_CONNECT
*Jan 29 21:32:51.909: r2_q421_ic_answer(0/2/0:1(1)) Tx ANSWER seizure: delay 0 ms,elapsed
32419 msvnm_dsp_set_sig_state:[R2 Q.421 0/2/0:1(1)] set signal state = 0x4
*Jan 29 21:32:51.910: r2_reg_channel_connected(0/2/0:1(1))
*Jan 29 21:32:51.910: r2_reg_process_event: [0/2/0:1(1), R2_REG_WAIT_FOR_CONNECT,
E_R2_REG_CONNECT(90)]
*Jan 29 21:32:51.910: r2_reg_connect(0/2/0:1(1))htsp_call_service_msghtsp_call_service_msg
 not EFXS (11)
```

This example shows the output for the **debug vtsp all** command.

```
(config-controller)#debug vtsp all
Log Buffer (4096 bytes)::S_R2_DIALING_COMP, event:E_VTSP_DIGIT_END]
*Jan 29 21:56:33.690: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/dc_digit:
*Jan 29 21:56:33.690: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_process_event:
[state:S_R2_DIALING_COMP, event:E_TSP_R2_DIAL]
*Jan 29 21:56:33.690: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/dc_dial:
*Jan 29 21:56:33.690: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dial_nopush:
*Jan 29 21:56:33.690: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/ds_do_dial:     Digits To
Dial=#
*Jan 29 21:56:33.901: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_dial_done_cb:
*Jan 29 21:56:33.901: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_process_event:
[state:S_R2_DIALING_COMP, event:E_VTSP_DSM_DIALING_COMPLETE]
*Jan 29 21:56:33.901: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/dc_dialing_done:
*Jan 29 21:56:34.690: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_process_event:
[state:S_R2_DIALING_COMP, event:E_TSP_R2_END_DIAL]
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/ds_end_dial:
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_digit_pop:
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_digit_pop:     Digit
Reporting=FALSE
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/act_alert_dial_complete:
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/act_service_msg_down:
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_timer_stop:     Timer
Stop Time=80497275
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_peer_event_cb:
Event=E_DSM_CC_CAPS_ACK
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/act_service_msg_down:
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_timer_stop:     Timer
Stop Time=80497275
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_peer_event_cb:
```

```
Event=E_DSM_CC_CAPS_ACK
*Jan 29 21:56:34.691: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_peer_event_cb:
Event=E_DSM_CC_CAPS_ACK
*Jan 29 21:56:34.692: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_feature_notify_cb:
    Feature ID=0, Feature Status=1
*Jan 29 21:56:34.692: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_reactivate_ringback:

*Jan 29 21:56:34.692:
//213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_reactivate_ringback:exit@1299
*Jan 29 21:56:34.693: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_feature_notify_cb:
    Feature ID=0, Feature Status=1
*Jan 29 21:56:34.693: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_reactivate_ringback:

*Jan 29 21:56:34.693:
//213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_reactivate_ringback:exit@1299
*Jan 29 21:56:34.693: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_feature_notify_cb:
    Feature ID=0, Feature Status=1
*Jan 29 21:56:34.693: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_reactivate_ringback:

*Jan 29 21:56:34.693:
//213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_reactivate_ringback:exit@1299
*Jan 29 21:56:58.140: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_call_connect: Connected
 Name
*Jan 29 21:56:58.140: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_call_connect: Connected
 Number 39701
*Jan 29 21:56:58.140: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_call_connect: Connected
 oct3a  30
*Jan 29 21:56:58.140: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_process_event:
[state:S_ALERTING, event:E_CC_CONNECT]
*Jan 29 21:56:58.140: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/act_alert_connect:    Progress
 Indication=2
*Jan 29 21:56:58.140: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_ring_noan_timer_stop:
   Timer Stop Time=80499620
*Jan 29 21:56:58.142: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_process_event:
[state:S_CONNECT, event:E_CC_SERVICE_MSG]
*Jan 29 21:56:58.142: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/act_service_msg_down:
*Jan 29 21:56:58.142: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_timer_stop:     Timer
Stop Time=80499620
*Jan 29 21:56:58.144: //213/85E8EDFC81D1/VTSP:(0/2/0:1):0:1:1/vtsp_dsm_fpi_event_cb:
Event=E_DSMP_FPI_ENABLE_TDM_RTCP
```

Feature Group D Configuration

To configure the Feature Group D signaling, perform these steps:

Before you begin

The Feature Group D signaling is supported on Cisco 4000 Series Integrated Services Routers from IOS XE release 15.5 (2). Feature Group D service is a trunk side connection that enables telephone customers to choose their long distance network and use the same number of digits irrespective of carrier they use. Routers interface with interexchange carriers using Feature Group D to support voice traffic in the carrier environment.

Before you attempt this configuration, ensure that you meet these prerequisites:

- The platform must be using Digital T1/E1 Packet Voice Trunk Network Modules.

- The Digital T1/E1 Packet Voice Trunk Network Module can have one or two slots for voice/WAN Interface Network Modules (NIMs); NIM supports one to eight ports. Only the dual-mode (voice/WAN) multiple trunk cards are supported in the digital E1 packet voice trunk network module, not older VICs.
- Drop-and-Insert capability is supported only between two ports on the same multiple card.

SUMMARY STEPS

1. **configure terminal** {*ip-address* | *interface-type* *interface-number* [*ip-address*]}
2. **voice-card** **slot/subslot**
3. **controller T1/E1** **slot/subslot/port**
4. **framing** {*sf* | *esf* }
5. **linecode** {*b8zs* | *ami*}
6. **ds0-group** *ds0-group-no***timeslots** `timeslot-list type`{*e&m-fgd* | *fgd-eana*}
7. **no shutdown**
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** {*ip-address* | *interface-type* *interface-number* [*ip-address*]} **Example:** `Router(config)# configure terminal`	Enters global configuration mode.
Step 2	**voice-card** **slot/subslot** **Example:** `Router(config)# voice-card slot/subslot`	Enters voice card interface configuration mode and specify the slot location by using a value from 0 to 5, depending upon your router.
Step 3	**controller T1/E1** **slot/subslot/port** **Example:** `Router(config)# controller T1 slot/subslot/port`	Enters controller configuration mode for the T1 controller at the specified slot/port location. Valid values for slot and port are 0 and 1.
Step 4	**framing** {*sf* | *esf* } **Example:** `Router(config)# framing {sf | esf}`	Sets the framing according to your service provider's instructions. Choose Extended Superframe (ESF) format or Superframe (SF) format.
Step 5	**linecode** {*b8zs* | *ami*}	Sets the line encoding according to your service provider's instructions. Bipolar-8 zero substitution (B8ZS) encodes a sequence of eight zeros in a unique binary sequence to detect line coding violations. Alternate mark inversion (AMI) represents zeros using a 01 during each bit cell, and ones are represented by 11 or 00, alternately, during each bit cell. AMI requires that the sending device maintain ones density.

	Command or Action	Purpose
		Ones density is not maintained independent of the data stream.
Step 6	**ds0-group** *ds0-group-no***timeslots** *timeslot-list* *type*{*e&m-fgd* | *fgd-eana*}	Defines the T1 channels for use by compressed voice calls as well as the signaling method the router uses to connect to the PBX or CO. ds0-group-no is a value from 0 to 23 that identifies the DS0 group. Note The ds0-group command automatically creates a logical voice port that is numbered as follows: slot/port:ds0-group-no. Although only one voice port is created, applicable calls are routed to any channel in the group. timeslot-list is a single number, numbers separated by commas, or a pair of numbers separated by a hyphen to indicate a range of timeslots. For T1, allowable values are from 1 to 24. To map individual DS0 timeslots, define additional groups. The system maps additional voice ports for each defined group. The signaling method selection for type depends on the connection that you are making. The e&m-fgd setting allows E&M interface connections for PBX trunk lines (tie lines) and telephone equipment to use feature group D switched-access service. The fgd-eana setting supports the exchange access North American (EANA) signaling.
Step 7	**no shutdown**	Activates the controller.
Step 8	**exit**	Exits controller configuration mode. Skip the next step if you are not setting up Drop and Insert .

Media and Signaling Authentication and Encryption

The Media and Signaling Authentication and Encryption Feature for Cisco IOS MGCP Gateways feature implements voice security features that include signaling authentication along with media and signaling encryption on MGCP gateways. For more information on Media and Signaling Authentication and Encryption Feature, see the http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/voice/mgcp/configuration/15-mt/vm-15-mt-book/vm-gw-med-sig.html

Multicast Music-on-Hold

The Music-on-Hold (MOH) feature enables you to subscribe to a music streaming service when you are using a Cisco IOS MGCP voice gateway. Music streams from an MOH server to the voice interfaces of on-net and off-net callers that have been placed on hold. Cisco Communications Manager supports the capability to place callers on hold with music supplied from a streaming multicast MOH server.

By means of a preconfigured multicast address on the Cisco Unified Communications Manager or gateway, the gateway can "listen" for Real-Time Transport Protocol (RTP) packets that are broadcast from a default router in the network and can relay the packets to designated voice interfaces in the network. You can initiate the call on hold. However, you cannot initiate music on hold on a MGCP controlled analog phone. Whenever a called party places a calling party on hold, Cisco Communications Manager requests the MOH server to stream RTP packets to the "on-hold" interface through the preconfigured multicast address. In this way, RTP packets are relayed to appropriately configured voice interfaces that have been placed on hold. When you configure a multicast address on a gateway, the gateway sends an Internet Gateway Management Protocol (IGMP) "join" message to the default router, indicating to the default router that the gateway is ready to receive RTP multicast packets.

Multiple MOH servers can be present in the same network, but each server must have a different Class D IP address, and the address must be configured in Cisco Communications Manager and the MGCP voice gateways. For more information on configuring MOH, see the http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/voice/cminterop/configuration/15-0m/vc-15-0m-book/vc-ucm-mgcp-gw.html#GUID-A3461142-2F05-4420-AEE6-032FCA3B7952

TLS 1.2 support on SCCP Gateways

The TLS 1.2 support on SCCP Gateways feature details the configuration of TLS 1.2 on SCCP protocol for digital signal processor (DSP) farm including Unicast conference bridge

(CFB), Media Termination Point (MTP), and SCCP telephony control (STC) application (STCAPP).

DSP on gateways can be used as media resources for transrating or transcoding. Each media resource uses Secure Skinny Client Control Protocol (SCCP) to communicate with Cisco Unified Communications Manager. Currently SSL 3.1, which is equivalent to TLS1.0, is used for sending secure signals. This feature enhances the support to TLS 1.2. From Cisco IOS XE Cupertino 17.7.1a, TLS 1.2 is enhanced to support the Next-Generation Encryption (NGE) cipher suites.

Note Cisco Unified Communications Manager (CUCM) Version 14SU2 has been enhanced to support Secured SCCP gateways with the Subject Name field (CN Name) with or without colons, for example, AA:22:BB:44:55 or AA22BB4455.

CUCM checks the CN field of the incoming certificate from the SCCP Gateway and verifies it against the DeviceName configured in CUCM for this gateway. DeviceName contains MAC address of the gateway. CUCM converts the MAC address in the DeviceName to MAC address with colons (for example: AA:22:BB:44:55) and validates with the CN name in the Gateway's certificate. Therefore, CUCM mandates Gateway to use MAC address with colons for the CN field in the certificate, that is, subject name.

Due to new guidelines from Defense Information Systems Agency (DISA), it is a requirement not to use colons for the subject name field CN. For example, AA22BB4455.

SCCP TLS connection

CiscoSSL is based on OpenSSL. SCCP uses CiscoSSL to secure the communication signals.

If a resource is configured in the secure mode, the SCCP application initiates a process to complete Transport Layer Security (TLS) handshaking. During the handshake, the server sends information to CiscoSSL about the TLS version and cipher suites supported. Previously, only SSL3.1 was supported for SCCP secure signalling. SSL3.1 is equivalent to TLS 1.0. The TLS 1.2 Support feature introduces TLS1.2 support to SCCP secure signalling.

After TLS handshaking is complete, SCCP is notified and SCCP kills the process.

If the handshaking is completed successfully, a REGISTER message is sent to Cisco Unified Communications Manager through the secure tunnel. If handshaking fails and a retry is needed, a new process is initiated.

Note For SCCP-based signalling, only TLS_RSA_WITH_AES_128_CBC_SHA cipher suite is supported.

Cipher Suites

For SCCP-based signaling, TLS_RSA_WITH_AES_128_CBC_SHA cipher suite is supported.

From Cisco IOS XE Cupertino 17.7.1a, the following NGE cipher suites are also supported:

- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-RSA-AES256-GCM-SHA384

These cipher suites enable secure voice signaling for both the STCAPP analog phone and the SCCP DSPFarm conferencing service. The cipher suite selection is negotiated between gateway and CUCM.

The following prerequisites are applicable for using NGE cipher suites:

- Configure TLS 1.2. For more information, see Configuring TLS version for STC application, on page 462.
- Use CUCM Release 14.1 SU1 or later, and Voice Gateways or platforms that support TLS 1.2.
- From the CUCM Web UI, navigate to **Cipher Management** and set the **CIPHER switch** as **NGE**. For more information, see Cipher Management.

For more information about verifying cipher suites, see Verifying TLS Version and Cipher Suites, on page 462.

For the SRTP-encrypted media, you can use higher-grade cipher suites - AEAD-AES-128-GCM or AEAD-AES-256-GCM. The selection of these cipher suites is automatically negotiated between GW and CUCM for both secure analog voice and hardware conference bridge voice media. Authenticated Encryption with Associated Data (AEAD) ciphers simultaneously provide confidentiality, integrity, and authenticity, without built-in SHA algorithms to validate message integrity.

Supported Platforms

The TLS 1.2 support on the SCCP Gateways feature is supported on the following platforms:

- Cisco 4321 Integrated Services Router
- Cisco 4331 Integrated Services Router
- Cisco 4351 Integrated Services Router
- Cisco 4431 Integrated Services Router
- Cisco 4451-X Integrated Services Router
- Cisco 4461 Integrated Services Router
- Cisco Catalyst 8200 and 8300 Series Edge Platforms
- Cisco VG400, VG420, and VG450 Analog Voice Gateways

Configuring TLS version for STC application

Perform the following task to configure a TLS version for the STC application:

```
enable
configure terminal
stcapp security tls-version v1.2
exit
```

Note The stcapp security tls command sets the TLS version to v.1.0, v1.1, or v1.2 only. If not configured explicitly, TLS v1.0 is selected by default.

Configuring TLS version in Secure Mode for DSP Farm Profile

Perform the following task to configure the TLS version in secure mode for DSP farm profile:

```
enable
configure terminal
dspfarm profile 7 conference security
  tls-version v1.2
  exit
```

Note Note: The **tls** command can be configured only in security mode.

Verifying TLS Version and Cipher Suites

Perform the following task to verify the TLS version and cipher suite:

```
# show dspfarm profile 100
Dspfarm Profile Configuration

 Profile ID = 100, Service = CONFERENCING, Resource ID = 2
 Profile Service Mode : secure
 Trustpoint : Overlord_DSPFarm_GW
 TLS Version  : v1.2
 TLS Cipher   : ECDHE-RSA-AES256-GCM-SHA384
 Profile Admin State : UP
 Profile Operation State : ACTIVE
 Application : SCCP   Status : ASSOCIATED
 Resource Provider : FLEX_DSPRM   Status : UP
 Total Number of Resources Configured : 10
 Total Number of Resources Available : 10
 Total Number of Resources Out of Service : 0
 Total Number of Resources Active : 0
 Maximum conference participants : 8
 Codec Configuration: num_of_codecs:6
 Codec : g711ulaw, Maximum Packetization Period : 30 , Transcoder: Not Required
 Codec : g711alaw, Maximum Packetization Period : 30 , Transcoder: Not Required
 Codec : g729ar8, Maximum Packetization Period : 60 , Transcoder: Not Required
 Codec : g729abr8, Maximum Packetization Period : 60 , Transcoder: Not Required
 Codec : g729r8, Maximum Packetization Period : 60 , Transcoder: Not Required
 Codec : g729br8, Maximum Packetization Period : 60 , Transcoder: Not Required
```

Verifying STCAPP Application TLS Version

Perform the following tasks to verify the TLS version of the STCAPP application:

```
Device# show call application voice stcapp
App Status: Active
CCM Status: UP
CCM Group: 120
Registration Mode: CCM
Total Devices: 0
Total Calls in Progress: 0
Total Call Legs in Use: 0
ROH Timeout: 45
TLS Version: v1.2

# show stcapp dev voice 0/1/0
Port Identifier:  0/1/0
Device Type:      ALG
Device Id:        585
Device Name:      ANB3176C85F0080
Device Security Mode : Encrypted
  TLS version        : TLS version 1.2
  TLS cipher         : ECDHE-RSA-AES256-GCM-SHA384
Modem Capability: None
Device State:     IS
Diagnostic:       None
Directory Number: 80010
Dial Peer(s):     100
Dialtone after remote onhook feature: activated
Busytone after remote onhook feature: not activated
Last Event:       STCAPP_CC_EV_CALL_MODIFY_DONE
Line State:       ACTIVE
Line Mode:        CALL_CONF
Hook State:       OFFHOOK
mwi:              DISABLE
vmwi:             OFF
mwi config:       Both
Privacy:          Not configured
HG Status:        Unknown
PLAR:             DISABLE
Callback State:   DISABLED
CWT Repetition Interval: 0 second(s) (no repetition)
Number of CCBs:   1
Global call info:
    Total CCB count      = 3
    Total call leg count = 6

Call State for Connection 2 (ACTIVE): TsConnected
Connected Call Info:
   Call Reference: 33535871
   Call ID (DSP):  187
   Local IP Addr:  172.19.155.8
   Local IP Port:  8234
   Remote IP Addr: 172.19.155.61
   Remote IP Port: 8154
   Calling Number: 80010
   Called Number:
   Codec:          g711ulaw
   SRTP:           on
   RX Cipher:      AEAD_AES_256_GCM
   TX Cipher:      AEAD_AES_256_GCM
```

Perform the following task to verify the sRTP cipher suite for the DSPfarm connection:

```
# show sccp connection detail

bridge-info(bid, cid) - Normal bridge information(Bridge id, Calleg id)
mmbridge-info(bid, cid) - Mixed mode bridge information(Bridge id, Calleg id)
```

```
sess_id    conn_id    call-id    codec   pkt-period dtmf_method    type
bridge-info(bid, cid)   mmbridge-info(bid, cid) srtp_cryptosuite         dscp
                      call_ref   spid       conn_id_tx

16778224   -          125        N/A     N/A        rfc2833_pthru     confmsp   All RTPSPI
 Callegs      All MM-MSP Callegs      N/A                        N/A
                      -          -          -

16778224   16777232   126        g711u   20         rfc2833_pthru  s- rtpspi    (101,125)
    N/A                              AEAD_AES_256_GCM          184
                      30751576   16777219   -

16778224   16777231   124        g711u   20         rfc2833_pthru  s- rtpspi    (100,125)
    N/A                              AEAD_AES_256_GCM          184
                      30751576   16777219   -


Total number of active session(s) 1, connection(s) 2, and callegs 3
```

Verifying Call Information

To display call information for TDM and IVR calls stored in the Forwarding Plane Interface (FPI), use the **showvoipfpi calls** command. You can select a call ID and verify the cipher suite using the **show voip fpi calls confID** *call_id_number* command. In this example, cipher suite 6 is AES_256_GCM.

```
#show voip fpi calls
Number of Calls : 2
---------- ---------- ---------- ----------- --------------- ---------------
    confID correlator     AcallID     BcallID           state           event
---------- ---------- ---------- ----------- --------------- ---------------
         1          1         87          88       ALLOCATED DETAIL_STAT_RSP
        21         21         89          90       ALLOCATED DETAIL_STAT_RSP

#show voip fpi calls confID 1
---------------------------------------------------------------------------
VoIP-FPI call entry details:
---------------------------------------------------------------------------
Call Type        :           TDM_IP     confID           :                1
correlator       :                1     call_state       :        ALLOCATED
last_event       :  DETAIL_STAT_RSP     alloc_start_time :       1796860810
modify_start_time:                0     delete_start_time:                0
Media Type(SideA):             SRTP     cipher suite     :                6
---------------------------------------------------------------------------
FPI State Machine Stats:
------------------------
create_req_call_entry_inserted             :         1
.........
```

Additional References

Related Topic	Document Title
Cisco IOS Voice Gateways Configuration Guide	Supplementary Services Features for FXS Ports on Cisco IOS Voice Gateways Configuration Guide

Feature Information for TLS 1.2 support on SCCP Gateways

Table 45: Feature Information for TLS 1.2 support on SCCP Gateways

Feature Name	Releases	Feature Information
TLS 1.2 support on SCCP Gateways	Cisco IOS XE Fuji 16.7.1	The TLS 1.2 support on SCCP Gateways feature details the configuration of TLS 1.2 on SCCP protocol for DSP farm including CFB, MTP, and STCAPP. The following commands were introduced: **stcapp security tls-version**, **tls-version**.
Support for NGE Cipher Suites	Cisco IOS XE Cupertino 17.7.1a	This feature supports NGE cipher suites for secure voice signaling and secure media. These cipher suites are applicable for both the STCAPP analog phone and the SCCP DSPFarm conferencing service.

CHAPTER 29

Dying Gasp Through SNMP, Syslog and Ethernet OAM

Dying Gasp—One of the following unrecoverable condition occurs:

- System reload
- Interface shutdown
- Power failure—supported on specific platforms

This type of condition is vendor specific. An Ethernet Operations, Administration, and Maintenance (OAM) notification about the condition may be sent immediately.

Prerequisites for Dying Gasp Support

You must enable Ethernet OAM before configuring Simple Network Management Protocol (SNMP) for dying gasp feature. For more information, see Enabling Ethernet OAM on an Interface.

Restrictions for Dying Gasp Support

- The native GigabitEthernet interfaces on the Cisco ISR 4000 platforms do not support generating dying-gasp SNMP traps in the following scenarios:
  - The router goes down after removal of the power supply unit (PSU).
  - The router goes down after removal of the power cable.
- The dying gasp support feature cannot be configured using CLI. To configure hosts using SNMP, refer to the SNMP host configuration examples below.

- In the case of system reload or interface shutdown on the Cisco 4000 Series ISRs and Cisco 1100 Series ISRs running Cisco IOS-XE Everest Release 16.6.2, dying gasp packets are sent to peer routers. However, the system state is not captured in the system logs (syslogs) or SNMP traps.

Information About Dying Gasp Through SNMP, Syslog and Ethernet OAM

Dying Gasp

One of the OAM features as defined by IEEE 802.3ah is Remote Failure Indication, which helps in detecting faults in Ethernet connectivity that are caused by slowly deteriorating quality. Ethernet OAM provides a mechanism for an OAM entity to convey these failure conditions to its peer via specific flags in the OAM PDU. One of the failure condition method to communicate is Dying Gasp, which indicates that an unrecoverable condition has occurred; for example, when an interface is shut down. This type of condition is vendor specific. A notification about the condition may be sent immediately and continuously.

How to Configure Dying Gasp Through SNMP, Syslog and Ethernet OAM

Dying Gasp Trap Support for Different SNMP Server Host/Port Configurations

Note You can configure up to five different SNMP server host/port configurations.

Environmental Settings on the Network Management Server

```
setenv SR_TRAP_TEST_PORT=UDP port
setenv SR_UTIL_COMMUNITY=public
setenv SR_UTIL_SNMP_VERSION=v2c
setenv SR_MGR_CONF_DIR=Path to the executable snmpinfo.DAT file
```

The following example shows SNMP trap configuration on the host:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
Router(config)# snmp-server host 10.0.0.149 vrf Mgmt-intf version 2c public udp-port 6264
Router(config)#
Router(config)# ^Z
Router#
```

After performing a power cycle, the following output is displayed on the router console:

```
Router#
System Bootstrap, Version 16.6(2r), RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1994-2017  by cisco Systems, Inc.
Current image running: Boot ROM0
Last reset cause: LocalSoft
C1111-8PLTELA platform with 4194304 Kbytes of main memory
rommon 1 >

=======================================
Dying Gasp Trap Received for the Power failure event:
-----------------------------------------------------
  Trap on the Host
++++++++++++++

snmp-server host = 10.0.0.149 (nms1-lnx)  and SR_TRAP_TEST_PORT=6264
/auto/sw/packages/snmpr/15.4.1.9/bin> /auto/sw/packages/snmpr/15.4.1.9/bin/traprcv
Waiting for traps.
Received SNMPv2c Trap:
Community: public
From: 10.29.25.101
snmpTrapOID.0 = ciscoMgmt.305.1.3.5.0.2
ciscoMgmt.305.1.3.6 = Dying Gasp - Shutdown due to power loss
```

Message Displayed on the Peer Router on Receiving Dying Gasp Notification

```
001689: *May 30 14:16:47.746 IST: %ETHERNET_OAM-6-RFI: The client on interface Gi0/0/0 has
 received a remote failure indication from its remote peer(failure reason = remote client
power failure action = )
```

Displaying SNMP Configuration for Receiving Dying Gasp Notification

Use the show running-config command to display the SNMP configuration for receiving dying gasp notification:

```
Router# show running-config | i snmp
snmp-server community public RW
snmp-server host 10.0.0.149 vrf Mgmt-intf version 2c public udp-port 6264
Router#
```

Configuration Examples for Dying Gasp Through SNMP, Syslog and Ethernet OAM

Example: Configuring SNMP Community Strings on a Router

Setting up the community access string to permit access to the SNMP:

```
Router> enable
Router# configure terminal
Router(config)# snmp-server community public RW
Router(config)# exit
```

For more information on command syntax and examples, refer to the Cisco IOS Network Management Command Reference.

Example: Configuring SNMP-Server Host Details on the Router Console

Specifying the recipient of a SNMP notification operation:

```
Router> enable
Router# configure terminal
Router(config)# snmp-server host X.X.X.XXX vrf mgmt-intf version 2c public udp-port 9800
Router(config)# exit
```

For more information on command syntax and examples, refer to the Cisco IOS Network Management Command Reference.

Feature Information for Dying Gasp Support

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 46: Feature Information for Dying Gasp Support

Feature Name	Releases	Feature Information
Dying Gasp	Cisco IOS XE Release 16.6.2	Ethernet OAM provides a mechanism for an OAM entity to convey failure conditions to its peer via specific flags in the OAM PDU. One of the failure condition method to communicate is Dying Gasp, which indicates that an unrecoverable condition has occurred; for example, when an interface is shut down. This type of condition is vendor specific. A notification about the condition may be sent immediately and continuously.

CHAPTER 30

Support for Software Media Termination Point

The Support for Software Media Termination Point (MTP) feature bridges the media streams between two connections, allowing Cisco Unified Communications Manager (CUCM) to relay the calls that are routed through SIP or H.323 endpoints through Skinny Client Control Protocol (SCCP) commands. These commands allow CUCM to establish an MTP for call signaling.

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 https://cfnng.cisco.com/. An account on Cisco.com is not required.

Information About Support for Software Media Termination Point

This feature extends the software MTP support to the Cisco Unified Border Element (Enterprise). Software MTP is an essential component of large-scale deployments of Cisco UCM. This feature enables new capabilities so that the Cisco UBE can function as an Enterprise Edge Cisco Session Border Controller for large-scale deployments that are moving to SIP trunking.

Prerequisites for Software Media Termination Point

- For the software MTP to function properly, codec and packetization must be configured the same way on both in call legs and out call legs.

Restrictions for Software Media Termination Point

- RSVP Agent is not supported in software MTP.
- Software MTP for repacketization is not supported.
- Call Threshold is not supported for standalone software MTP.
- Per-call debugging is not supported.
- Multiple concurrent Synchronisation Sources (SSRCs) with the same destination IP and port are not supported.

SRTP-DTMF Interworking

From Cisco IOS XE 17.10.1a, Secure Real-time Transport Protocol (SRTP) Dual-Tone Multi-Frequency (DTMF) interworking is supported with Software MTP in pass through mode. SMTP supports DTMF Interworking for nonsecure calls, and this feature adds support for SRTP DTMF interworking for secure calls.

CUCM support for this feature is expected to be implemented in a later release.

Restrictions for SRTP-DTMF Interworking

- The SRTP-DTMF Interworking feature supports only the codec-passthrough format.
- The SRTP-DTMF Interworking feature does not support multiple concurrent Synchronised Sources (SSRCs) with the same destination IP and port.
- The calls that support SRTP-DTMF Interworking may have a minor performance impact as compared to calls supported on nonsecure DTMF interworking.

Supported Platforms for SRTP-DTMF Interworking

From Cisco IOS XE 17.10.1a, the following platforms support SRTP DTMF interworking with SMTP:

- Cisco 4461 Integrated Services Router (ISR)
- Cisco Catalyst 8200 Edge Series Platforms
- Cisco Catalyst 8300 Edge Series Platforms
- Cisco Catalyst 8000V Edge Software

Configuring Support for Software Media Termination Point

Perform the following tasks to enable and configure the support for Software Media Termination Point feature.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **sccp local** *interface-type interface-number* [**port** *port-number*]

4. **sccp ccm** {*ipv4-address* | *ipv6-address* | *dns*} **identifier** *identifier-number* [**port** *port-number*] **version** *version-number*
5. **sccp**
6. **sccp ccm group** *group-number*
7. **associate ccm** *identifier-number* **priority** *number*
8. **associate profile** *profile-identifier* **register** *device-name*
9. **dspfarm profile** *profile-identifier* {**conference** | **mtp** | **transcode**} [**security**]
10. **trustpoint** *trustpoint-label*
11. **codec** *codec*
12. **maximum sessions** {**hardware** | **software**} *number*
13. **associate application sccp**
14. **no shutdown**

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	**sccp local** *interface-type interface-number* [**port** *port-number*] **Example:** `Router(config)# sccp local gigabitethernet0/0/0`	Selects the local interface that SCCP applications (transcoding and conferencing) use to register with Cisco UCM. • *interface type*: Can be an interface address or a virtual-interface address such as Ethernet. • *interface number*: Interface number that the SCCP application uses to register with Cisco UCM. • (Optional) **port** *port-number*: Port number used by the selected interface. Range is 1025 to 65535. Default is 2000.
Step 4	**sccp ccm** {*ipv4-address* | *ipv6-address* | *dns*} **identifier** *identifier-number* [**port** *port-number*] **version** *version-number* **Example:** `Router(config)# sccp ccm 10.1.1.1 identifier 1 version 7.0+`	Adds a Cisco UCM server to the list of available servers and sets the following parameters: • *ipv4-address*: IP version 4 address of the Cisco UCM server. • *ipv6-address*: IP version 6 address of the Cisco UCM server. • *dns*: DNS name.

	Command or Action	Purpose
		• **identifier**: Specifies the number that identifies the Cisco UCM server. Range is 1 to 65535. • **port** *port-number* (Optional): Specifies the TCP port number. Range is 1025 to 65535. Default is 2000. • **version** *version-number*: Cisco UCM version. Valid versions are 3.0, 3.1, 3.2, 3.3, 4.0, 4.1, 5.0.1, 6.0, and 7.0+. There is no default value.
Step 5	**sccp** **Example:** `Router(config)# sccp`	Enables the Skinny Client Control Protocol (SCCP) and its related applications (transcoding and conferencing).
Step 6	**sccp ccm group** *group-number* **Example:** `Router(config)# sccp ccm group 10`	Creates a Cisco UCM group and enters SCCP Cisco UCM configuration mode. • *group-number*: Identifies the Cisco UCM group. Range is 1 to 50.
Step 7	**associate ccm** *identifier-number* **priority** *number* **Example:** `Router(config-sccp-ccm)# associate ccm 10 priority 3`	Associates a Cisco UCM with a Cisco UCM group and establishes its priority within the group: • *identifier-number*: Identifies the Cisco UCM. Range is 1 to 65535. There is no default value. • **priority** *number*: Priority of the Cisco UCM within the Cisco UCM group. Range is 1 to 4. There is no default value. The highest priority is 1.
Step 8	**associate profile** *profile-identifier* **register** *device-name* **Example:** `Router(config-sccp-ccm)# associate profile 1 register MTP0011`	Associates a DSP farm profile with a Cisco UCM group: • *profile-identifier*: Identifies the DSP farm profile. Range is 1 to 65535. There is no default value. • **register** *device-name*: Device name in Cisco UCM. A maximum of 15 characters can be entered for the device name.
Step 9	**dspfarm profile** *profile-identifier* {**conference** | **mtp** | **transcode**} [**security**] **Example:** `Router(config-sccp-ccm)# dspfarm profile 1 mtp`	Enters DSP farm profile configuration mode and defines a profile for DSP farm services: • *profile-identifier*: Number that uniquely identifies a profile. Range is 1 to 65535. There is no default. • **conference**: Enables a profile for conferencing. • **mtp**: Enables a profile for MTP. • **transcode**: Enables a profile for transcoding.

	Command or Action	Purpose
		• **security**(Optional): Enables a profile for secure DSP farm services. For more information on configuration examples, see section #unique_497 unique_497_Connect_42_GUID-5FB6A48E-204C-45AA-AE63-413B075A7871, on page 477.
Step 10	**trustpoint** *trustpoint-label* **Example:** `Router(config-dspfarm-profile)# trustpoint dspfarm`	(Optional) Associates a trustpoint with a DSP farm profile.
Step 11	**codec** *codec* **Example:** `Router(config-dspfarm-profile)# codec g711ulaw`	Specifies the codecs supported by a DSP farm profile. • codec-type: Specifies the preferred codec. Enter ? for a list of supported codecs Repeat this step for each supported codec.
Step 12	**maximum sessions** {**hardware** | **software**} *number* **Example:** `Router(config-dspfarm-profile)# maximum sessions software 10`	Specifies the maximum number of sessions that are supported by the profile. • **hardware**: Number of sessions that MTP hardware resources can support. • **software**: Number of sessions that MTP software resources can support. • *number*: Number of sessions that are supported by the profile. Range is 0 to x. Default is 0. The x value is determined at run time depending on the number of resources available with the resource provider.
Step 13	**associate application sccp** **Example:** `Router(config-dspfarm-profile)# associate application sccp`	Associates SCCP to the DSP farm profile.
Step 14	**no shutdown** **Example:** `Router(config-dspfarm-profile)# no shutdown`	Changes the status of the interface to the UP state.

Examples: Support for Software Media Termination Point

The following example shows a sample configuration for the Support for Software Media Termination Point feature:

```
sccp local GigabitEthernet0/0/1
```

```
sccp ccm 10.13.40.148 identifier 1 version 6.0
sccp
!
sccp ccm group 1
 bind interface GigabitEthernet0/0/1
 associate ccm 1 priority 1
 associate profile 6 register RR_RLS6
!
 dspfarm profile 6 mtp
 codec g711ulaw
 maximum sessions software 100
 associate application SCCP
!
!
gateway
media-inactivity-criteria all
timer receive-rtp 400
```

The following example shows a sample configuration for the SRTP-DTMF Interworking feature-with secure dspfarm profile:

```
sccp local GigabitEthernet0/0/0
sccp ccm 172.18.151.125 identifier 1 version 7.0
sccp
!
sccp ccm group 1
 bind interface GigabitEthernet0/0/0
 associate ccm 1 priority 1
 associate profile 1 register Router
!
dspfarm profile 1 mtp security
 trustpoint IOSCA
 codec g711ulaw
 codec pass-through
 tls-version v1.2
 maximum sessions software 5000
 associate application SCCP
```

Note SR-TP traffic can pass through an SMTP resource when the dspfarm profile is provisioned with codec pass-through, and if it does not have TLS and security-related configuration. For traffic flows that require SRTP-DTMF interworking support, the SMTP dspfarm profile must include the **security** keyword and the TLS and codec pass-through configuration. This dspfarm resource profile can also pass through SRTP traffic independent of SRTP-DTMF interworking support.

Verifying Software Media Termination Point Configuration

To verify and troubleshoot this feature, use the following **show** commands.

- To verify information about SCCP, use the **show sccp** command:

```
Router# show sccp

SCCP Admin State: UP
Gateway IP Address: 10.13.40.157, Port Number: 2000
IP Precedence: 5
User Masked Codec list: None
```

```
Call Manager: 10.13.40.148, Port Number: 2000
                Priority: N/A, Version: 6.0, Identifier: 1
                Trustpoint: N/A
```

- To verify information about the DSPfarm profile, use the **show dspfarm profile** command:

```
Router# show dspfarm profile 6

Dspfarm Profile Configuration
 Profile ID = 6, Service = MTP, Resource ID = 1
 Profile Description :
 Profile Service Mode : Non Secure
 Profile Admin State : UP
 Profile Operation State : ACTIVE
 Application : SCCP   Status : ASSOCIATED
 Resource Provider : NONE   Status : NONE
 Number of Resource Configured : 100
 Number of Resource Available : 100
 Hardware Configured Resources : 0
 Hardware Available Resources : 0
 Software Resources : 100
 Codec Configuration
 Codec : g711ulaw, Maximum Packetization Period : 30
```

- To verify information about the secure DSPfarm profile status, use the **show dspfarm profile** command and check that the secure service mode is set:

```
Router# show dspfarm profile 2
Dspfarm Profile Configuration
 Profile ID = 2, Service = MTP, Resource ID = 2
 Profile Service Mode : secure
 Trustpoint : IOSCA
 TLS Version  : v1.2
 TLS Cipher   : AES128-SHA
 Profile Admin State : UP
 Profile Operation State : ACTIVE
 Application : SCCP   Status : ASSOCIATED
 Resource Provider : NONE   Status : NONE
 Total Number of Resources Configured : 8000
 Total Number of Resources Available : 8000
 Total Number of Resources Out of Service : 0
 Total Number of Resources Active : 0
 Hardware Configured Resources : 0
 Hardware Resources Out of Service: 0
 Software Configured Resources : 8000
 Number of Hardware Resources Active : 0
 Number of Software Resources Active : 0
 Codec Configuration: num_of_codecs:2
 Codec : pass-through, Maximum Packetization Period : 0
 Codec : g711ulaw, Maximum Packetization Period : 30
```

- To display statistics for the SCCP connections, use the **show sccp connections** command:

```
Router# show sccp connections

sess_id   conn_id    stype  mode       codec    ripaddr        rport   sport
16808048  16789079    mtp   sendrecv   g711u    10.13.40.20    17510   7242
16808048  16789078    mtp   sendrecv   g711u    10.13.40.157   6900    18050
```

For SMTP secure DTMF, the **show sccp connections** command displays the codec type (pass-th), the s-type (s-mtp), and information about the DTMF method (rfc2833_pthru):

```
Router# show sccp connections

sess_id   conn_id   stype   mode     codec    sport  rport   ripaddr conn_id_tx   dtmf_method
16791234  16777308  s-mtp   sendrecv  pass_th  8006   24610   172.18.153.37
rfc2833_pthru
16791234  16777306  s-mtp   sendrecv  pass_th  8004   17576   172.18.154.2
rfc2833_report

Total number of active session(s) 1, and connection(s) 2
```

- To display information about RTP connections, use the **show rtpspi call** command:

```
Router# show rtpspi call
RTP Service Provider info:
No. CallId  dstCallId   Mode      LocalRTP  RmtRTP  LocalIP     RemoteIP    SRTP
1    22      19         Snd-Rcv   7242      17510   0x90D080F   0x90D0814    0
2    19      22         Snd-Rcv   18050     6900    0x90D080F   0x90D080F    0
```

If SRTP DTMF interworking is active, the SRTP field shows a non-zero value:

```
Router# show rtpspi call
RTP Service Provider info:
No. CallId  dstCallId   Mode      LocalRTP  RmtRTP  LocalIP     RemoteIP    SRTP
1   13      14          Snd-Rcv   8024      18270   0xA7A5355   0xAC129A02   1
2   14      13          Snd-Rcv   8026      24768   0xA7A5355   0xAC129925   1
```

- To display information about VoIP RTP connections, use the **show voip rtp connections** command:

```
Router# show voip rtp connections
VoIP RTP Port Usage Information
Max Ports Available: 30000, Ports Reserved: 100, Ports in Use: 102
Port range not configured, Min: 5500, Max: 65499
VoIP RTP active connections :
No. CallId    dstCallId  LocalRTP  RmtRTP    LocalIP          RemoteIP
1     114       117        19822    24556   10.13.40.157     10.13.40.157
2     115       116        24556    19822   10.13.40.157     10.13.40.157
3     116       115        19176    52625   10.13.40.157     10.13.40.20
4     117       114        16526    52624   10.13.40.157     10.13.40.20
```

- Additional, more specific, **show** commands that can be used include the following:
    - **show sccp connection callid**
    - **show sccp connection connid**
    - **show sccp connection sessionid**
    - **show rtpspi call callid**
    - **show rtpspi stat callid**
    - **show voip rtp connection callid**
    - **show voip rtp connection type**
    - **show platform hardware qfp active feature sbc global**
- To isolate specific problems, use the **debug sccp** command:
    - **debug sccp** [**all** | **config** | **errors** | **events** | **keepalive** | **messages** | **packets** | **parser** | **tls**]

Feature Information for Support for Software Media Termination Point

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 47: Feature Information for Support for Software Media Termination Point

Feature Name	Releases	Feature Information
Support for Software Media Termination Point	Cisco IOS XE Release 2.6 S	Software Media Termination Point (MTP) provides the capability for Cisco Unified Communications Manager (Cisco UCM) to interact with a voice gateway via Skinny Client Control Protocol (SCCP) commands. These commands allow the Cisco UCM to establish an MTP for call signaling.
Support for Secure Real-time Transport Protocol (SRTP) Dual-Tone Multi-Frequency (DTMF) Interworking	Cisco IOS XE Dublin 17.10.1a	The Secure Real-time Transport Protocol (SRTP) Dual-Tone Multi-Frequency (DTMF) feature provides support for DTMF interworking between Secure Software MTP in pass-through mode only and CUCM.

CHAPTER 31

LTE Support on Cisco 4000 Series Integrated Services Router

This chapter provides an overview of the software features and configuration information for Cisco NIM LTE modules on the Cisco 4000 Series Integrated Services Router (ISR).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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

Overview of Cisco LTE

Note The LTE support feature is supported on Cisco 4000 Series Integrated Services Router (ISR) via Network Interface Modules (NIMs). For more information on the list of NIMs for ISR 4K, please see Interfaces and Modules.

Cisco LTE supports the following modes:

- **4G LTE** —4G LTE mobile specification provides multi-megabit bandwidth, more efficient radio network, latency reduction, and improved mobility. LTE solutions target new cellular networks. These networks initially support up to 300 Mb/s peak rates in the downlink and up to 50 Mb/s peak rates in the uplink. The throughput of these networks is higher than the existing 3G networks.
- **3G Evolution High-Speed Packet Access (HSPA/HSPA+)** —HSPA is a UMTS-based 3G network. It supports High-Speed Downlink Packet Access (HSDPA) and High-Speed Uplink Packet Access (HSUPA) data for improved download and upload speeds. Evolution High-Speed Packet Access (HSPA+) supports Multiple Input/Multiple Output (MIMO) antenna capability.

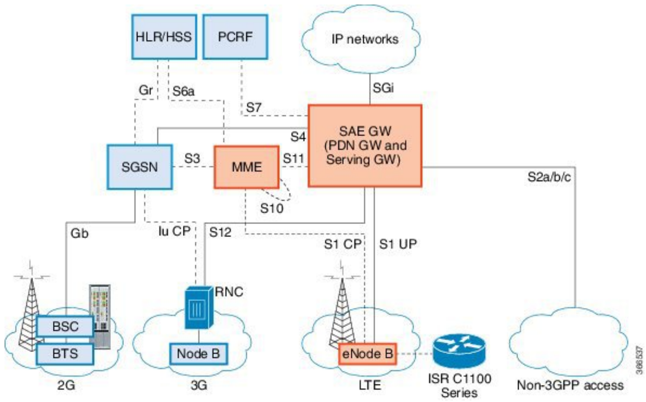
TThe following table describes the Cisco NIM LTE NIM-LTEA-EA and NIM-LTEA-LA SKUs:

Table 48: Cisco NIM LTE NIM-LTEA-EA and NIM LTEA-LA SKUs

Region Theaters	Cisco LTE Advanced 3.0 LTEEA SKU (European Union, North America)	Cisco LTE Advanced 3.0 LTELA SKUs (Latin America, Australia, Japan, China, India, Southeast Asia and South Korea)
Bands	LTE bands 1-5, 7, 12, 13, 20, 25, 26, 29, 30, and 41 FDD LTE 700 MHz (band 12), 700 MHz (band 29), 800 MHz (band 20), 850 MHz (band 5 CLR), 850 MHz (band 26 Low), 900 MHz (band 8), 1800 MHz (band 3), 1900 MHz (band 2), 1900 MHz (PCS band 25), 1700 MHz and 2100 MHz (band 4 AWS), 2100 MHz (band 1), 2300 MHz (band 30), or 2600 MHz (band 7) TDD LTE 2500 MHz (band 41) Carrier aggregation band combinations: 1+8; 2+(2,5,12,13,29); 3+(7,20); 4+(4,5,12,13,29); 7+(7,20); 12+30, 5+30, and 41+41	LTE bands 1, 3, 5, 7, 8, 18, 19, 21, 28, 38, 39, 40, and 41 FDD LTE 700 MHz (band 28), 850 MHz (band 5 CLR), 850 MHz (bands 18 and 19 Low), 900 MHz (band 8), 1500 MHz (band 21), 1800 MHz (band 3), 2100 MHz (band 1), or 2600 MHz (band 7) TDD LTE 1900 MHz (band 39), 2300 MHz (band 40), 2500 MHz (band 41), or 2600 MHz (band 38) Carrier aggregation band combinations: 1+(8,18,19,21); 3+(5,7,19,28); 7+(5,7,28); 19+21, 38+38, 39+39,40+40, and 41+41

The following figure explains the 4G LTE packet core network architecture.

Figure 4: 4G LTE Packet Core Network Architecture



Gateways	The Serving Gateway (SGW) routes and forwards user data packets, while also acting as the mobility anchor for the user plane, and is the anchor for mobility between LTE and other 3GPP technologies. The Packet Data Network (PDN) Gateway (PGW) provides connectivity from the User Equipment (UE) to external packet data networks by being the point of exit and entry of traffic for the UE. A UE may have simultaneous connectivity with more than one PGW for accessing multiple PDNs. The PGW performs policy enforcement, packet filtering for each user, charging support, lawful interception, and packet screening. Another key role of the PGW is to act as the anchor for mobility between 3GPP and non-3GPP technologies such as WiMAX and 3GPP2 (CDMA 1X and EvDO). The System Architecture Evolution GW (SAE GW) is the entity that covers the PGW and SGW functionality in the Evolved Packet Core (EPC).
RNC	The Radio Network Controller (RNC) is responsible for controlling the Radio Access Network (RAN) that are connected to it. The RNC carries out radio resource management and some of the mobility management functions and is the point where encryption is done before user data is sent to and from the mobile. The RNC connects to the Circuit-Switched Core Network through the Media Gateway (MGW).
BTS	Base Transceiver Station.
BSC	Base Station Controller.
SGSN	Service GPRS Support Node.

Prerequisites for Configuring Cisco LTE Support

- If the signal is not good at the router, use the Cisco offered antenna accessories and extension cables to place the antenna away from router in a better coverage area.
- You must have LTE Support network coverage where your router is physically placed. For a complete list of supported carriers.
- You must subscribe to a service plan with a wireless service provider and obtain a Subscriber Identity Module (SIM) card. Only micro SIM is supported.

- You must install the SIM card before configuring the LTE Support on Cisco Cisco ISR 4000 series router.
- The standalone antenna that supports GPS capabilities must be installed for the GPS feature to work. See the Cisco 4G Indoor/Outdoor Active GPS Antenna (GPS-ACT-ANTM-SMA) document for installation information.

Restrictions for Configuring Cisco LTE Support

- Currently, cellular networks support only user initiated bearer establishment.
- Due to the shared nature of wireless communications, the experienced throughput varies depending on the number of active users or congestion in a given network.
- Cellular networks have higher latency compared to wired networks. Latency rates depend on the technology and carrier. Latency also depends on the signal conditions and can be higher because of network congestion.
- CDMA-EVDO, CDMA-1xRTT, and GPRS technology modes are not supported.
- Any restrictions that are part of the terms of service from your carrier.
- SMS—Only one text message up to 160 characters to one recipient at a time is supported. Larger texts are automatically truncated to the proper size before being sent.
- It is strongly recommended that you configure SNMP V3 with authentication/privacy.

Features not Supported in Cisco LTE Support

The following features are not supported on Cisco LTE Support Cisco 4000 Series ISR:

- TTY support or Line
- Chat script/dialer string
- External Dialer
- DM log output to USB flash is not supported.

Cisco LTE Support Features

Cisco LTE Support supports the following major features:

- Global Positioning System (GPS) and National Marine Electronics Association (NMEA) streaming.
- Short Message Service (SMS)
- 3G/4G Simple Network Management Protocol (SNMP) MIB
- SIM lock and unlock capabilities
- Dual SIM

- Auto SIM
- NeMo
- Public Land Mobile Network (PLMN) selection
- IPv6
- Multiple PDN
- LTE Link Recovery

The following sections explains the Cisco LTE Support features:

4G GPS and NMEA

Active GPS is supported on the SubMiniature version A (SMA) port. Active GPS antenna is supported only in the standalone mode. An Active GPS antenna includes a built-in Low-Noise Amplifier that provides sufficient gain to overcome coaxial cable losses while providing the proper signal level to the GPS receiver. Active GPS antennae require power from the GPS receiver SMA port to operate. See the Example: Connecting to a Server Hosting a GPS Application, on page 487 for more information.

National Marine Electronics Association (NMEA) streams GPS data either from a LTE Support through a virtual COM port and a TCP/IP Ethernet connection to any marine device (such as a Windows-based PC) that runs a commercially available GPS-based application.

The following GPS and NMEA features are supported on the Cisco LTE Support:

- GPS standalone mode (satellite-based GPS)
- Cisco IOS CLI display coordinates.
- External application displays router map location
- Objects in the CISCO-WAN-3G-MIB supports GPS and NMEA features
- The Cisco LTE Support only supports NMEA over IP and uses show commands in the platform

Note Assisted GPS mode is not supported.

For instructions on setting up the GPS antenna, see the Cisco 4G Indoor/Outdoor Active GPS Antenna (GPS-ACT-ANTM-SMA) document.

Example: Connecting to a Server Hosting a GPS Application

You can feed the NMEA data to a remote server that hosts the GPS application. The server can be connected to the router either directly using an Ethernet cable or through a LAN or WAN network. If the application supports serial port, run a serial port emulation program to create a virtual serial port over the LAN or WAN connection.

Note Microsoft Streets & Trips is a licensed software that you can download from the Microsoft website.

To connect a Cisco LTE Support through IP to a PC running Microsoft Streets & Trips, perform the following steps:

1. Connect the PC to the router using an Ethernet cable.
2. Ensure that the PC and router can ping.
3. Launch the serial port redirector on the PC.
4. Create a virtual serial port that connects to the NMEA port on the router.
5. Launch **Microsoft Streets & Trips** on your PC.
6. Select the GPS Menu.
7. Click Start Tracking.
8. If you have acquired a location fix from the **show cellular 0/2/0 gps** command output on the router, the current location is plotted on the graph, and a reddish brown dotted cursor with a circle around it is seen on the map.

Note If you have not acquired a location fix, the Microsoft application times out and disconnects.

Dual SIM Card

SIM card primary slot is selected when router boots up or when NIM reloads. The default slot is 0. If SIM card is not present in the primary slot, select the alternative slot if SIM card is present.

```
controller cellular 0/2/0
lte sim primary slot <slot#>
```

If the active SIM card loses connectivity to the network a failover to the alternative SIM card slot occurs.

By default the failover timer is two minutes. The failover timer can be set from 1 to 7 minutes.

```
controller cellular 0/2/0
lte failovertimer <3-7>
```

You can also manually switch the SIM slot via the command line interface.

```
cellular 0/2/0 lte sim activate slot <0-1>
```

Auto SIM

The Auto SIM feature detects the SIM and loads the corresponding firmware. For example, if a Verizon SIM is detected, the modem loads the Verizon firmware. If you switch the SIM to an ATT SIM, the modem will load ATT firmware.

When Auto-SIM is enabled, it is said to be in Auto-SIM mode and when disabled, it is known as Manual mode. In Auto-SIM mode, the modem selects the right carrier firmware from the list of firmware's available. When in manual mode, you can select the firmware manually. Modem resets every time you make a config change from Auto-SIM enabled to disabled or vice-versa.

Note Auto SIM is always enabled by default.

Enable Auto SIM

SUMMARY STEPS

1. **Cellular** *slots/sub-slots/interface* **lte firmware-activate** *firmware-index*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Cellular** *slots/sub-slots/interface* **lte firmware-activate** *firmware-index* **Example:** `Router(config)# Cellular 0/2/0 lte firmware-activate 1`	Activates the firmware index. **Note** For the LTE Support, the *unit* argument identifies the slot, subslot, and the interface separated by slashes (0/2/0).

Example: List the firmware when Auto-SIM is Enabled

```
Device# show cellular 0/2/0 firmware
firmware          Idx Carrier      FwVersion      PriVersion   Status
 1   ATT          192.0.2.1        002.035_000  Inactive
 2   GENERIC      192.0.2.2        002.035_000  Active
 3   ROGERS       192.0.2.3        001.012_000  Inactive
 4   SPRINT       192.0.2.4        002.012_000  Inactive
 5   VERIZON      192.0.2.5        002.042_000  Inactive

Firmware Activation mode  =  AUTO
```

Disable Auto SIM

SUMMARY STEPS

1. **configure terminal**
2. **controller cellular** *slots/sub-slots/interface*
3. **no lte firmware auto-sim**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.

	Command or Action	Purpose
Step 2	**controller cellular** *slots/sub-slots/interface* **Example:** `Router(config)# controller cellular 0/2/0`	Specifies the controller interface.
Step 3	**no lte firmware auto-sim** **Example:** `Router(config-if)# no lte firmware auto-sim`	Disable auto SIM.

Example: List the firmware when Auto-SIM is Disabled

```
Device# show cellular 0/2/0 firmware
 Idx Carrier      FwVersion    PriVersion   Status
 1   ATT          192.0.2.1  002.035_000  Active
 2   GENERIC      192.0.2.2  002.035_000  Inactive
 3   ROGERS       192.0.2.3  001.012_000  Inactive
 4   SPRINT       192.0.2.4  002.012_000  Inactive
 5   VERIZON      192.0.2.5  002.042_000  Inactive


Firmware Activation mode  =  Manual
```

Using a SIM Card

Cisco LTE Support needs an active SIM card provided by a service provider. The SIM cards are usually provided in an unlocked state so that it can be used without a Personal Identification Number (PIN). If the SIM is unlocked, it can be inserted into a LTE Support and used without an authorization code.

The SIM can be initially locked with a PIN code (4 to 8 digits s long) defined by the service provider. Contact your service provider for the PIN code.

The SIM-Lock feature allows a SIM to be locked or unlocked with a PIN code so that it is used only in an authorized device. Perform the SIM lock and unlock procedures using the Cisco IOS CLI through a console or Telnet/SSH to the ISR.

After the SIM is locked, it cannot initiate a call unless authentication is done using the same PIN. Authentication is done automatically by Cisco IOS through configuration of the PIN. This mandatory configuration for automatic SIM authentication is done using the Cisco IOS CLI as part of the router startup configuration.

After the Cisco IOS configuration is in place, the ISR can initiate an LTE connection. The ISR uses the configured PIN to authenticate prior to the LTE connection. If the Cisco IOS PIN configuration is missing or if the PIN is incorrect, the SIM authentication will fail and the connection will not be initiated.

If the locked SIM is moved to a different ISR or to another device, or if the LTE in which the locked SIM resides is moved to a different LTE Support slot in the same ISR, the ISR configuration should be changed. The configuration is associated with the cellular controller that is specific to an ISR LTE slot number. This will ensure that the SIM card will not be used in any unauthorized device, or, if there are multiple LTE in a single ISR, that the appropriate PIN is applied to each LTE SIM. An authentication command (with the same

PIN used to lock the SIM) must be defined on the new device or on the new cellular controller slot to successfully initiate the LTE connection.

The following procedures are used to configure a SIM:

Caution It is very important to use the correct PIN after it is configured. The SIM card will be blocked if the wrong PIN is entered three consecutive times on a locked SIM during authentication or when trying to unlock a locked SIM. You can unblock a blocked SIM card using the PUK code. Contact your service provider for the PUK code. Use the **cellular** *<slot>* **lte sim unblock** *<PUK code> <new PIN code>* command to unblock the SIM.

Changing the PIN

Ensure to enter the correct PIN, the SIM card gets blocked if the wrong PIN is entered three consecutive times.

SUMMARY STEPS

1. **cellular** *slots subslots interface* **lte sim change-pin** *current-pin new-pin*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**cellular** *slots subslots interface* **lte sim change-pin** *current-pin new-pin* **Example:** `Router# cellular 0/2/0 lte sim lock 1111 1234`	Locks or unlocks the SIM card using a PIN code. **Note** Locks or unlocks the SIM card using a PIN code. *pin*—A code (4 to 8 digits long) provided by your service provider to lock or unlock the SIM card. **Note** SIM should be in locked state when the PIN is being changed.

Locking and Unlocking a SIM Card Using a PIN

Perform this task to lock or unlock a SIM card given by your service provider. Make sure you enter the correct PIN, the SIM card gets blocked if the wrong PIN is entered three consecutive times.

Procedure

	Command or Action	Purpose
Step 1	**cellular** *unit* **lte sim** {**lock** | **unlock**} *pin* **Example:** `Router# cellular 0/2/0 lte sim lock 1111`	Locks or unlocks the SIM card using a PIN code. **Note** *pin*—A code (4 to 8 digits long) provided by your service provider to lock or unlock the SIM card.

Configure CHV1 for Unencrypted Level 0

Procedure

	Command or Action	Purpose
Step 1	**cellular** *slots subslots interface* **lte sim lte sim authenticate 0 pin** **Example:** `Router# controller cellular 0/0/0`	Enters the cellular controller configuration mode Use either of these commands:**lte sim authenticate 0 pin** or **lte sim authenticate 0 pin slot {0 | 1}**

Configure CHV1 for Unencrypted Level7

To configure an encrypted PIN, the scrambled value of the PIN must be obtained. To get the scrambled Level 7 PIN and to configure the SIM CHV1 code for verification using this encrypted PIN, enter the following commands in the EXEC mode. When obtaining the encrypted PIN for a SIM, a username and password are created by configuring password encryption, defining the username and associated password, copying the resulting scrambled password, and using this scrambled password in the SIM authentication command.

Note After the scrambled PIN has been obtained and used in SIM authentication, the username created can be deleted from the Cisco IOS configuration. A SIM should be locked for SIM authentication to work.

Procedure

	Command or Action	Purpose
Step 1	**service password-encryption** **Example:** `Router (config)# service password-encryption`	Enables password encryption.
Step 2	username <*username*> **privilege** *var* password <*pin*> **Example:** `Router (config)# username SIM privilege 0 password 1111`	**Note** Creates username and password. name - specifies the username *pin*—A 4 to 8 digits PIN code.
Step 3	**do show run | i name** **Example:** `Device(config)# do show run | i SIM`	Shows the username configuration line with the encrypted level 7 PIN for the username created in Step 3 (user “SIM” in the example shown). Copy the scrambled password for use in Step 6 (as the PIN).
Step 4	*username* **privilege 0** *password pin* **Example:** `Device(config)# controller cellular 0/0/0`	Enters the cellular controller configuration mode.

	Command or Action	Purpose
Step 5	**lte sim authenticate 7** *pin* `OR`**lte sim authenticate 7 pin slot {0 | 1}** **Example:** `Device(config-controller)# lte sim authenticate 7 055A575E70`	Authenticates the SIM CHV1 code by using the encrypted keyword 7 and the scrambled PIN from Step 4. The PIN is sent to the modem for authentication with each subsequent LTE connection. If authentication passes based on the configured PIN, the data call is allowed. If authentication fails, the modem does not initiate the data call. **Note** The slot keyword and its options are available only on platforms that supports Dual-SIM feature.
Step 6	**exit** **Example:** `Device(config-controller)# exit`	(Optional) Exits the cellular controller configuration mode.
Step 7	**no username***name* **Example:** `Device(config-controller)# no username SIM`	(Optional) Removes the username and password created in Step 3
Step 8	**no service password-encryption***name* **Example:** `Device(config-controller)# no service password-encryption`	(Optional) Removes the username and password created in Step 3

Short Message Service (SMS) Capabilities

Cisco LTE Support receiving, transmitting, archiving, and deleting of SMS messages. This support includes the ability to view up to 25 received texts, and archive more messages in a custom file location. SMS is supported on multiple carriers. Cisco LTE Support also have the capability to revert from LTE SMS to 3G and 2G SMS technology if necessary.

A sending device behind a Cisco LTE Support transmits an SMS text message over the 4G cellular link through cellular towers until it the message reaches the recipient's router, which then notifies the recipient device, such as a cell phone. The receiving device uses the same process to return a reply to the sending device. The following figure describes the flow from a mobile device to a sending device. For SMS transmission to work, end users must have a text-capable device, and optionally, a text plan. If end users do not have a text plan, standard SMS rates apply to their text transmissions.

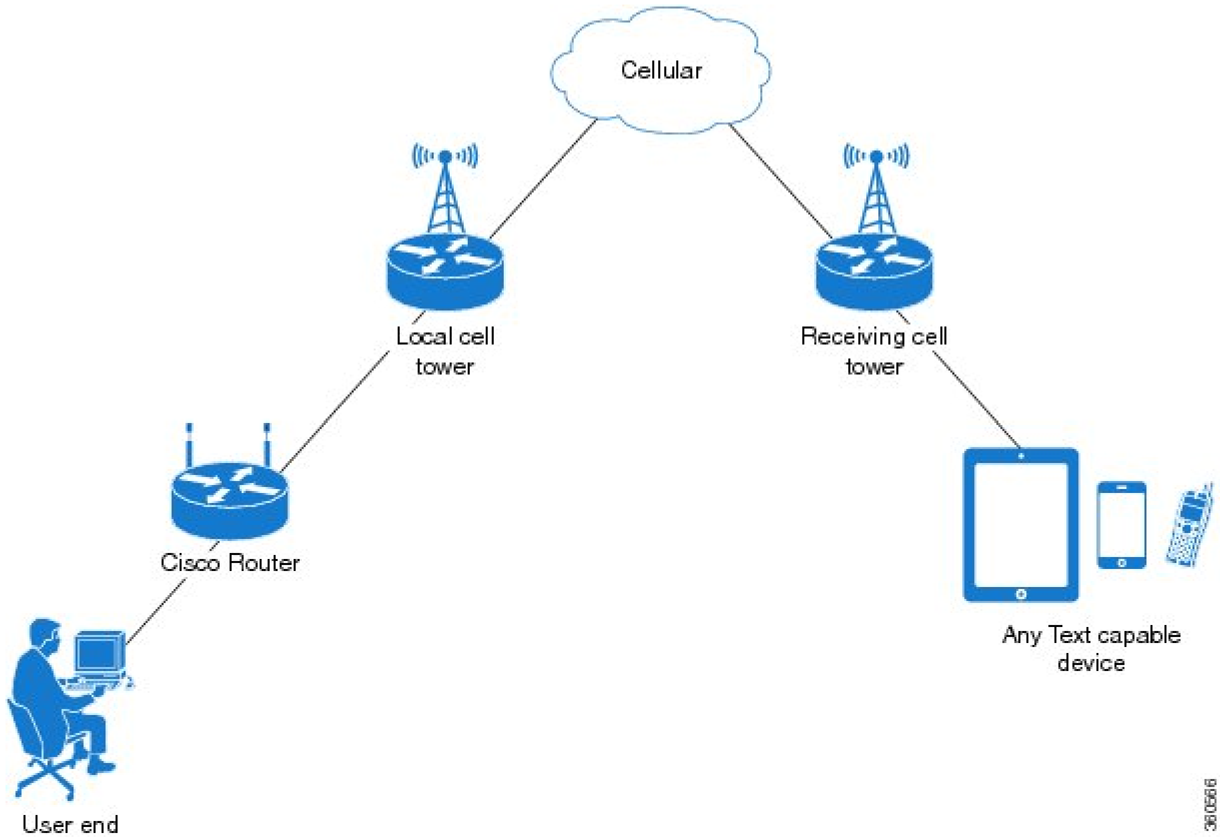


Figure 5: SMS Network

Data Account Provisioning

One or more modem data profiles can be created to provision a modem on a LTE SKU. An active wireless account with a service provider with one or more (dual) SIM cards must be installed. The modem data profile is pre-configured on the modem.

The following tasks are used to verify the signal strength and service availability of the modem and to create, modify, and delete modem data profiles:

IP Multimedia Subsystem Profiles

IP Multimedia Subsystem (IMS) profiles establish a session, and are a part of the modem configuration and are stored in the modem's NVRAM. An IMS network is an access-independent and standard-based IP connectivity service that enables different types of multimedia services to end users using common Internet-based protocols.

LTE LEDs

The following table describes the LED behavior in LTE.

Table 49: LTE LED Indicators

LED	Color/Bar and Description	
LTE SIM(0) & SIM(1)	Green (Solid)	Modem up, SIM installed and active
	Green Blink	LTE data activity
	Off	Modem not up; or modem up and no SIM
	Amber (Solid)	Modem up, SIM installed but not active
RSSI - Uses Bars for LED Indication	Four Bar	High RSSI >= -69dBm
	Three Bar	Medium RSSI, -89dBm <> -70dBm
	Two Bar	Low RSSI, -99dBm <> -90dBm
	One Bar	RSSI <= -100dBm
	0 or No Bar	No Service
SERVICE - Uses Color Indication	Green(solid)	LTE signal present (RSSI LEDs will be Green)
	Amber(solid)	2G/3G signal present (RSSI LEDs will be Amber)
	No Color	No service detected.
GPS	Green (Solid)	GPS coordinates are obtained.
	Off	GPS is disabled, GPS is enabled without GPS mode and NMEA configuration, or GPS is acquiring

Configuring Cisco LTE

For LTE, the numbering for slot 0, module 0, and port 0 is 0/2/0 for all commands.

Verifying Modem Signal Strength and Service Availability

For the LTE, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

Procedure

	Command or Action	Purpose
Step 1	**show cellular** *unit* **network** **Example:** `Router# show cellular 0/2/0 network`	Displays information about the carrier network, cell site, and available service.
Step 2	**show cellular** *unit* **radio** **Example:** `Router# show cellular 0/2/0 radio`	Shows the radio signal strength. **Note** The RSSI should be better than –90 dBm for steady and reliable connection.
Step 3	**show cellular** *unit* **profile** **Example:** `Router# show cellular 0/2/0 profile`	Shows information about the modem data profiles created.
Step 4	**show cellular** *unit* **security** **Example:** `Router# show cellular 0/2/0 security`	Shows the security information for the modem, such as SIM and modem lock status.
Step 5	**show cellular** *unit* **all** **Example:** `Router# show cellular 0/2/0 all`	Shows consolidated information about the modem, profiles created, radio signal strength, network security, and so on.

Guidelines for Creating, Modifying, or Deleting Modem Data Profiles

Customized profiles (Access Point Name (APN) in mobile networks) can be created and used on Cisco LTE SKU's. Maximum number of profiles that can be created are 16.

Cisco SKU's shipping with specific carrier provisioning file (Can be found in Carrier label under "show cellular <slot> hardware"), default profiles are already populated and can be deployed readily.

In all other cases where profile configurations are not available, separate profiles should be created with required parameters.

You can create multiple profiles on Cisco LTE. The following are the default internet profile numbers for the modems:

NIM SKU	Profile Number
NIM-LTEA-EA	Profile 1
NIM-LTEA-LA	Both Profile 1 and Profile 3

Follow these guidelines when you configure a data profile using EXEC mode or Config mode :

- You do not have to make any profile-related changes if your modem comes with a data profile, for instance, AT&T, Sprint and Verizon.
- If any profile parameter changes are required for a connection type, the changes will likely be carried out in the default profiles.
- To configure different profile types and use them for a different connection, you can create separate profiles with different parameters (for instance, APN names). Note that only one profile is active at a given time.
- Use the **show cellular <unit> profile** command to view the data profile. An asterisk(*) symbol is displayed against the data profile. Double asterisk(**) symbol is displayed against the attach profile.
- The data profile is used to set up a data call. If you want to use a different profile, that profile needs to be made the default one. Use the **lte sim data-profile** *number* command to change the default profile under **controller cellular 0/2/0**.

Creating, Modifying, or Deleting Data Profiles Using EXEC Mode

Customized profiles (Access Point Name (APN) in mobile networks) can be created and used on Cisco LTE SKU's. Maximum number of profiles that can be created are 16.

Cisco SKU's shipping with specific carrier provisioning file (can be found in carrier label under **show cellular** *slot* **hardware**, default profiles are already populated and can be deployed readily.

Note For the LTE, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

Procedure

	Command or Action	Purpose
Step 1	**cellular** *unit* **lte profile** *[***create** | **delete***] profile-number [apn [authentication [username password [bearer-type]]]]* **Example:** `Router# cellular 0/2/0 lte profile create 2 apn.com pap username pwd ipv4`	Creates, modifies, or deletes a modem data profile in the privileged EXEC mode. • The *profile-number* argument specifies the profile number created for the modem. • (Optional) The *apn* argument specifies an Access Point Name (APN). An APN is provided by your service provider. Only a single APN can be specified for a single profile. • (Optional) The *authentication* parameter specifies the authentication type used. Acceptable parameters are **chap**, **none** (no authentication), **pap**, and **pap_chap** (PAP or CHAP authentication). • (Optional) The *username* and *password* arguments are given by a service provider. These are mandatory when an authentication type other than **none** is used. • (Optional) The *PDN* type parameter specifies the type of packet data session established with mobile network

	Command or Action	Purpose
		using this profile. Acceptable parameters are: **ipv4** **ipv6** and **ipv4v6** (IPv4 and IPv6). The **show cellular** *slot* profile displays configured profile list. **Note** Single asterisk(*) displayed against data profile. Double asterisk(**) displayed against attached profile.

Example

```
router# show cellular 0/2/0 profile
Profile 1 = INACTIVE **
--------
PDP Type = IPv4v6
Access Point Name (APN) = vzwims
Authentication = None

Profile 2 = INACTIVE
--------
PDP Type = IPv4v6
Access Point Name (APN) = vzwadmin
Authentication = None

Profile 3 = ACTIVE*
--------
PDP Type = IPv4v6
PDP address = 192.0.2.1
PDP IPV6 address = 2600:1010:B00E:1E11:192D:3E20:199B:3A70/64  Scope: Global
Access Point Name (APN) = VZWINTERNET
Authentication = None
        Primary DNS address = 192.0.2.2
        Secondary DNS address = 192.0.2.2
        Primary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
        Secondary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
```

Note If data and attach profile bindings need modification, use the **controller cellular** slot.

```
router(config-controller)# lte sim data-profile 3 attach-profile 2 slot unit


Device #show cellular 0/2/0 profile
Profile 1 = INACTIVE
--------------------------------------------------
PDP Type = IPv4v6
Access Point Name (APN) = test
Authentication = None

Profile 2 = INACTIVE **
--------
PDP Type = IPv4
Access Point Name (APN) = internet
```

```
Authentication = PAP or CHAP
Username =  user@solution.com
Password =  cisco

Profile 3 = INACTIVE*
--------
PDP Type = IPv4v6
Access Point Name (APN) = basic
Authentication = None

  * - Default profile
 ** - LTE attach profile
Configured default profile for active SIM 0 is profile 2.
```

Creating, Modifying, or Deleting Data Profiles in Configuration Mode

> **Note** For the LTE NIM, the *unit* argument identifies the router slot, WIC slot, and port separated by slashes (0/1/0).

Procedure

	Command or Action	Purpose

Step 1	**profile id***id* **apn** *apn name [***authentication** *[username password]***pdn-type** *[pdn type][***slot***slot-number| no-overwrite]]]]*
	
	Example:
	
	`Router(config-controller)# profile id 1 apn apn_internet authentication none pdn-type ipv4 slot 0`
		Configures a cellular profile in the configuration mode.
		
		• The *id* argument specifies the profile number created for the modem. The maximum number of profiles that can be created for each modem are given as follows:
		  • EM7455 – Up to 16 profiles
		  • EM7430 – Up to 16 profiles
		
		• (Optional) The *apn* argument specifies an Access Point Name (APN) in the profile. An APN is provided by your service provider. Only a single APN can be specified in a single profile.
		
		• (Optional) The *authentication* parameter specifies the authentication type used. Acceptable parameters are **chap**, **none** (no authentication), **pap**, and **pap_chap** (PAP or CHAP authentication).
		
		• (Optional) The *username* and *password* arguments are provided by a service provider. These are mandatory when an authentication type is used other than none.
		
		• (Optional) The *PDN-type* parameter specifies the type of packet data session established with mobile network using this profile. Acceptable parameters are: **ipv4**, **ipv6** and **ipv4v6**.
		
		• (Optional) The *slot-number* parameter specifies the slot number. By default, the slot-number is the current active slot-number, if not specified.

	Command or Action	Purpose
		• (Optional) *No-overwrite* action to be taken when a profile already exists in modem for the profile id. If there is a profile already exists in the modem for this profile id and no-overwrite option is specified, this configuration will not overwrite existing profile. Default is *overwrite*.

Configuration Examples

The following example shows how to change a default profile on LTE:

```
router(config-controller)# lte sim data-profile 2 attach-profile 1 slot <unit>
```

The following example shows the output of the **show cellular** command for Verizon network service:

```
router# show cellular 0/2/0 profile
Profile 1 = INACTIVE **
--------
PDP Type = IPv4v6
Access Point Name (APN) = vzwims
Authentication = None

Profile 2 = INACTIVE
--------
PDP Type = IPv4v6
Access Point Name (APN) = vzwadmin
Authentication = None

Profile 3 = ACTIVE*
--------
PDP Type = IPv4v6
PDP address = 192.0.2.1
PDP IPV6 address = 2600:1010:B00E:1E11:192D:3E20:199B:3A70/64  Scope: Global
Access Point Name (APN) = VZWINTERNET
Authentication = None
        Primary DNS address = 192.0.2.2
        Secondary DNS address = 192.0.2.3
        Primary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
        Secondary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF

Profile 4 = INACTIVE
--------
PDP Type = IPv4v6
Access Point Name (APN) = vzwapp
Authentication = None

Profile 5 = INACTIVE
--------
PDP Type = IPv4v6
Access Point Name (APN) = vzw800
Authentication = None

Profile 6 = INACTIVE
--------
PDP Type = IPv4v6
Access Point Name (APN) = CISCO.GW4.VZWENTP
Authentication = None
```

```
  * - Default profile
 ** - LTE attach profile
```

Configuration Example

Example Configuration under Controller Cellular

```
router(config-controller)# profile id 1 apn apn_internet authentication none pdn-type ipv4
 no-overwrite
```

Controller Cellular Running Configuration

```
Router #show running-config controller cellular <slot>
Building configuration...

Current configuration : 330 bytes
!
controller Cellular 0/2/0
profile id 1 apn apn_internet authentication none pdn-type ipv4 no-overwrite
end



 ** This will override exec mode profile configuration
 ** If for a profile ID, configuration CLI exists, exec mode configuration cannot be
performed.
 Router #show cellular <slot> profile 5
Profile 5 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = apn_old
Authentication = None

TSN1#cellular <slot> lte profile create 5 apn_new
 Warning: You are attempting to create Profile 5
 Profile 5 was configured through controller configuration 'profile id <profile #>'
 Please execute command under controller configuration using '[no] profile id <profile #>'
 for profile 5 to create
 Profile 5 NOT written to modem


** As part of this enhancement, any attach and/or data profile changes will immediately
trigger a connection reset and take effect. Below warning message will be displayed.

Warning: You are attempting to modify the data/attach profile.
Connection will be reset
```

Configure Radio Band Selection

This feature allow users to configure and lock down the modem to a specific RF band, or set of bands. The preference can be set to be equal to, or a sub-set of the capability supported by the modem/carrier combination.

The following examples show the controller configuration commands.

:

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`Device# conf t`
	`Enter configuration commands, one per line.  End with CNTL/Z.`
		Enters configuration mode.

Step 2	**controllercellular***interface-number* **Example:** `Device(config)# controller cellular 0/2/0`	Configures the cellular interface on a network controller. The interface number is used to identify the specific interface being configured.
Step 3	**lte modem band-selectindicesumts3g***indices***lte4g***indices*[**nr5g***indices*]**slot***slot #* **Example:** `Device(config-controller)# lte modem band-select indices umts3g 24 lte4g 48 nr5g 40 slot 0`	Allows the user to choose frequency bands for their LTE modem, UMTS 3G, LTE 4G networks and for a specific SIM slot.

Example

```
router#show cellular 0/2/0 radio ?
  band     Show Radio band settings
  history  Show Radio history in graph format
  |        Output modifiers
  <cr>     <cr>

router#show cell 0/2/0 radio band
LTE bands supported by modem:
- Bands 1 2 3 4 5 7 8 12 13 14 18 19 20 25 26 28 29 30 32 34 38 39 40 41 42 43 46 48 66 71.
LTE band Preference settings for the active sim(slot 0):
- Bands 1 2 3 4 5 7 8 12 13 14 18 19 20 25 26 28 29 30 32 34 38 39 40 41 42 43 46 48 66 71.


NR5G bands supported by modem:
- Bands 1 2 3 5 7 8 12 20 25 28 38 40 41 48 66 71 77 78 79.
NR5G band Preference settings for the active sim(slot 0):
- Bands 1 2 3 5 7 8 12 20 25 28 38 40 41 48 66 71 77 78 79.

3G bands supported by modem:
Index: <none>
3G band Preference settings for the active sim(slot 0):
Index: <none>

===========================================

Band index reference list:

For LTE and 5G, indices 1-128 correspond to bands 1-128.

For 3G, indices 1-64 maps to the 3G bands mentioned against each above.
```

Multiple PDN Contexts

This feature enables router to connect to multiple (currently two) packet data networks. This allows users to enable different features independently on each PDN. For instance, the first PDN can be used for public Internet access and the second one for VPN connectivity; each PDN has its own set of IP addresses and QoS characteristics.

During the initialization of the router, two cellular interfaces corresponding to the two PDNs are created:

cellular 0/2/0 and cellular 0/2/1

These interfaces can be viewed as two logical interfaces using the same radio resources.

The interface cellular 0/2/0 is referred as the first PDN, and cellular 0/2/1 as the second PDN.

To bring up the two PDNs, configuration needs to be applied on both the cellular interfaces in order to make two simultaneous data calls. The next step is to associate the data-bearer profile with its corresponding cellular interface or PDN. It is sufficient to associate the profile for just the first PDN under the controller cellular configuration. Note that the second PDN assumes a profile that is just one above the profile used for the first PDN. For example, if the first PDN uses profile 1, the second PDN uses profile 2 automatically when the call is initiated for the second one.

After the interesting traffic is routed through these cellular interfaces, data calls are initiated and each interface is assigned its own IP and DNS addresses provided by the cellular network.

Note Both PDNs share radio resources. Therefore, any throughput measurement needs to take into account the aggregate throughput on both PDNs, instead of just one.

Note For Verizon cellular network, the second PDN uses profile #6 automatically, when the call is initiated for the second data connection.

Configuration Examples

The following example shows how to configure multiple PDN on Cisco LTE SKU:

```
interface Cellular0/2/0
ip address negotiated
 dialer in-band
 dialer idle-timeout 0
 dialer-group 1
 ipv6 enable
 pulse-time 1
!
interface Cellular0/2/1
ip address negotiated
dialer in-band
dialer idle-timeout 0
dialer-group 1
ipv6 enable
pulse-time 1
! dialer-list 1 protocol ipv6 permit
!

ip route 192.0.2.1 255.255.255.0 Cellular0/2/0
```

```
ip route 192.0.2.2 255.255.255.255 Cellular0/2/1
!
```

The following show commands can be used to verify the status of the multiple PDN calls:

```
Router#sh cellular 0/2/0 profile
Profile 1 = ACTIVE* **
--------
PDP Type = IPv4v6
PDP address = 192.0.2.1
PDP IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF/64  Scope: Global
Access Point Name (APN) = broadband
Authentication = None
        Primary DNS address = 192.0.2.2
        Secondary DNS address = 192.0.2.3
        Primary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
        Secondary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF

.
.
.

Profile 16 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = broadband
Authentication = CHAP
Username: ipv4v6
Password: xxxxxx

  * - Default profile
 ** - LTE attach profile

Configured default profile for active SIM 0 is profile 1.

Router# sh cellular 0/2/0 connection
Profile 1, Packet Session Status = ACTIVE
        Cellular0/2/0:
        Data Packets Transmitted = 9 ,  Received = 9
        Data Transmitted = 900 bytes, Received = 900 bytes
        IP address = 192.0.2.1
        IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF/64  Scope: Global
        Primary DNS address = 192.0.2.2
        Secondary DNS address = 192.0.2.3
        Primary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
        Secondary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
Profile 2, Packet Session Status = ACTIVE
        Cellular0/2/1:
        Data Packets Transmitted = 7 ,  Received = 2
        Data Transmitted = 700 bytes, Received = 176 bytes
        IP address = 192.0.2.4
        IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF/64  Scope: Global
        Primary DNS address = 171.70.168.183
        Secondary DNS address = 192.0.2.5
        Primary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
        Secondary DNS IPV6 address = 2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
.
.
.
Profile 16, Packet Session Status = INACTIVE

Router#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   192.0.2.1       YES manual up                    up
GigabitEthernet0/0/1   unassigned      YES unset  administratively down down
```

```
GigabitEthernet0/1/0   unassigned      YES unset  administratively down down
GigabitEthernet0/1/1   unassigned      YES unset  administratively down down
GigabitEthernet0/1/2   unassigned      YES unset  administratively down down
GigabitEthernet0/1/3   unassigned      YES u
nset  administratively down down
GigabitEthernet0/1/4   unassigned      YES unset  administratively down down
GigabitEthernet0/1/5   unassigned      YES unset  administratively down down
GigabitEthernet0/1/6   unassigned      YES unset  administratively down down
GigabitEthernet0/1/7   unassigned      YES unset  administratively down down
Wl0/1/8                unassigned      YES unset  administratively down down
Cellular0/2/0          192.0.2.2      YES IPCP   up                    up
Cellular0/2/1          192.0.2.3    YES IPCP   up                    up
Vlan1                  unassigned      YES manual up                    down

Router#
Router# show ip dns view
DNS View default parameters:
DNS Resolver settings:
  Domain lookup is enabled
  Default domain name:
  Domain search list:
  Domain name-servers:
    192.0.2.1
    2001:4860:4860::8888
    192.0.2.2
    2001:DB8:0000:FFFF:FFFF:FFFF:FFFF:FFFF
    192.0.2.3
    8.8.8.8
DNS Server settings:
  Forwarding of queries is enabled
  Forwarder addresses: DNS View default parameters: DNS Resolver settings:
Domain lookup is enabled Default domain name: Domain search list: Domain name-servers:
192.0.2.1
192.0.2.2
192.0.2.3
DNS Server settings:
Forwarding of queries is enabled
Forwarder addresses:
Router#
```

Configuring a SIM for Data Calls

Locking and Unlocking a SIM Card Using a PIN Code

Perform this task to lock or unlock a SIM card given by your service provider.

The SIM card gets blocked if the wrong PIN is entered three consecutive times. Make sure you enter the correct PIN the SIM is configured with. If your SIM card gets blocked, contact your service provider for a PUK code. Using the PUK code, you can unblock the SIM card.

For LTE, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

Procedure

	Command or Action	Purpose
Step 1	**cellular** *unit* **lte sim** {**lock** | **unlock**} *pin* **Example:**	Locks or unlocks the SIM card using a PIN code. • *pin*—A code (4 to 8 digits long) provided by your carrier to lock or unlock the SIM card.

	Command or Action	Purpose
	`Router# cellular 0/2/0 lte sim lock 1111`	

Changing the PIN Code

Perform this task to change the PIN code of a SIM.

For LTE, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

Procedure

	Command or Action	Purpose
Step 1	**cellular** *unit* **lte sim change-pin** *pin new-pin* **Example:** `Router# cellular 0/2/0 lte sim change-pin 1111 1234`	Changes the assigned PIN code. SIM should be in locked state when the PIN is being changed.

Verifying the Security Information of a Modem

Perform this task to verify the security information of a modem.

Note For LTE, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

Procedure

	Command or Action	Purpose
Step 1	**show cellular** *unit* **security** **Example:** `Router# show cellular 0/2/0 security`	Shows the security information of the modem, including the SIM lock status.

Configuring Automatic Authentication for a Locked SIM

An unencrypted PIN can be configured to activate the Card Holder Verification (CHV1) code that authenticates a modem.

The SIM card gets blocked if the wrong PIN is entered three consecutive times. Make sure you enter the correct PIN the SIM is configured with. If your SIM card gets blocked, contact your service provider for a PUK code.

Follow these procedures when using an unencrypted Level 0 PIN to configure CHV1. For instructions on how to configure CHV1 using an encrypted Level 7 PIN, see the Configuring an Encrypted PIN for a SIM, on page 507.

A SIM should be locked for SIM authentication to work. To verify the SIM's status, use the **show cellular** *unit* **security** command.

For LTE, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**controller cellular** *unit* **Example:** `Router(config)# controller cellular 0/2/0`	Enters the cellular controller configuration mode.
Step 3	**lte sim authenticate 0** *pin*	Authenticates the SIM CHV1 code by using an unencrypted (**0**) keyword and PIN. This PIN is sent to the modem for authentication with each subsequent LTE connection. If authentication passes based on the configured PIN, the data call is allowed. If authentication fails, the modem does not initiate the data call. **Note** This command is valid only when an unencrypted PIN is used. To configure CHV1 code using an encrypted PIN, see the Configuring an Encrypted PIN for a SIM, on page 507.

Configuring an Encrypted PIN for a SIM

To configure an encrypted PIN, the scrambled value of the PIN must be obtained. To get the scrambled Level 7 PIN and to configure the SIM CHV1 code for verification using this encrypted PIN, enter the following commands in the EXEC mode.

Note When obtaining the encrypted PIN for a SIM, a username and password are created by configuring password encryption, defining the username and associated password, copying the resulting scrambled password, and using this scrambled password in the SIM authentication command. After the scrambled PIN has been obtained and used in SIM authentication, the username created can be deleted from the Cisco IOS configuration.

Note A SIM should be locked for SIM authentication to work. To verify the SIM's status, use the **show cellular** <*unit*> **security** command.

Note For the 4G LTE SKU, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

SUMMARY STEPS

1. **configure terminal**
2. **service password-encryption**
3. **username** *name* **privilege 0 password** *pin*
4. **do show run** | **i** *name*
5. **controller cellular** *unit*
6. **lte sim authenticate** {**0** | **7**} *pin*
7. **exit**
8. **no username** *name*
9. **no service password-encryption**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**service password-encryption** **Example:** `Router(config)# service password-encryption`	Enables password encryption.
Step 3	**username** *name* **privilege 0 password** *pin* **Example:** `Router(config)# username SIM privilege 0 password 1111`	Creates username and password. • *name*—Specifies the username. • *pin*—Specifies the four- to eight-digit PIN code.
Step 4	**do show run** | **i** *name* **Example:** `Router(config)# do show run | i SIM`	Shows the username configuration line with the encrypted level 7 PIN for the username created in Step 3 (user “SIM” in the example shown). Copy the scrambled password for use in Step 6 (as the PIN).
Step 5	**controller cellular** *unit* **Example:** `Router(config)# controller cellular 0/2/0`	Enters the cellular controller configuration mode.
Step 6	**lte sim authenticate** {**0** | **7**} *pin*	Authenticates the SIM CHV1 code by using the encrypted keyword **7** and the scrambled PIN from Step 4. The PIN is sent to the modem for authentication with each subsequent LTE connection. If authentication passes based on the configured PIN, the data call is allowed. If authentication fails, the modem does not initiate the data call.

	Command or Action	Purpose
Step 7	**exit** **Example:** `Router(config-controller)# exit`	(Optional) Exits the cellular controller configuration mode.
Step 8	**no username** *name* **Example:** `Router(config)# no username SIM`	(Optional) Removes the username and password created in Step 3.
Step 9	**no service password-encryption** **Example:** `Router(config)# no service password-encryption`	(Optional) Disables password encryption.

Applying a Modem Profile in a SIM Configuration

SUMMARY STEPS

1. **configure terminal**
2. **controller cellular** *unit*
3. **lte sim data-profile** *number* **attach-profile** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters the global configuration mode.
Step 2	**controller cellular** *unit* **Example:** `Router(config)# controller cellular 0/2/0`	Enters the cellular controller configuration mode.
Step 3	**lte sim data-profile** *number* **attach-profile** *number*	Applies the configured profile number to the SIM and its slot number. The default (primary) slot is 0. The **attach profile** is the profile used by the modem to attach to the LTE network. The **data profile** is the profile used to send and receive data over the cellular network.

Data Call Setup

To set up a data call, use the following procedures:

Configuring the Cellular Interface

To configure the cellular interface, enter the following commands starting in EXEC mode.

For the LTE, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

If a tunnel interface is configured with **ip unnumbered cellular 0/2/0**, it is necessary to configure the actual static IP address under the cellular interface, in place of **ip address negotiated**.

SUMMARY STEPS

1. **configure terminal**
2. **interface cellular** unit
3. **ip address negotiated**
4. **dialer in-band**
5. **dialer-group** group-number
6. **exit**
7. **ip route** *network-number network-mask* {*ip-address* | *interface*} [*administrative distance*] [**name** *name*]
8. **dialer-list** dialer-group **protocol** protocol-name {**permit** | **deny** | **list** *access-list-number* | **access-group**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**interface cellular** unit **Example:** `Router(config)# interface cellular 0/2/0`	Specifies the cellular interface.
Step 3	**ip address negotiated** **Example:** `Router(config-if)# ip address negotiated`	Specifies that the IP address for a particular interface is dynamically obtained.
Step 4	**dialer in-band** **Example:** `Router(config-if)# dialer in-band`	Enables DDR and configures the specified serial interface to use in-band dialing.
Step 5	**dialer-group** group-number **Example:**	Specifies the number of the dialer access group to which the specific interface belongs.

	Command or Action	Purpose
	`Router(config-if)# dialer-group 1`	
Step 6	**exit** **Example:** `Router(config-if)# exit`	Enters the global configuration mode.
Step 7	**ip route** *network-number network-mask* {*ip-address* | *interface*} [*administrative distance*] [**name** *name*] **Example:** `Router(config)# ip route 209.165.200.225 255.255.255.224 cellular 0/2/0`	Establishes a floating static route with the configured administrative distance through the specified interface. **Note** A higher administrative distance should be configured for the route through the backup interface so that it is used only when the primary interface is down.
Step 8	**dialer-list** dialer-group **protocol** protocol-name {**permit** | **deny** | **list** *access-list-number* | **access-group**} **Example:** `Router(config)# dialer-list 1 protocol ip list 1`	Creates a dialer list for traffic of interest and permits access to an entire protocol.

Configuring DDR

To configure DDR for the cellular interface, enter the following commands starting in EXEC mode.

Note For the LTE, the *unit* argument identifies the router slot, module slot, and port separated by slashes (0/2/0).

SUMMARY STEPS

1. **configure terminal**
2. **interface cellular** *unit*
3. **ip address negotiated**
4. **dialer in-band**
5. **ip address negotiated**
6. **dialer idle-timeout** *seconds*
7. dialer-group group-number
8. **exit**
9. dialer-list dialer-group protocol protocol-name {permit | deny | list *access-list-number* | access-group}
10. access-list access-list-number permit *ip*-source-address

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**interface cellular** *unit* **Example:** `Router(config)# interface cellular 0/2/0`	Specifies the cellular interface.
Step 3	**ip address negotiated** **Example:** `Router(config-if)# ip address negotiated`	Specifies that the IP address for a particular interface is dynamically obtained.
Step 4	**dialer in-band** **Example:** `Router(config-if)# dialer in-band`	Enables DDR and configures the specified serial interface to use in-band dialing.
Step 5	**ip address negotiated** **Example:** `Router(config-if)# ip address negotiated`	Specifies that the IP address for a particular interface is dynamically obtained.
Step 6	**dialer idle-timeout** *seconds* **Example:** `Router(config-if)# dialer idle-timeout 30`	Specifies the duration of idle time, in seconds, after which a line has no outbound traffic. "0" second means no idle timeout. The default idle timeout is 120 seconds if there is no idle timer specified.
Step 7	dialer-group group-number **Example:** `Router(config-if)# dialer-group 1`	Specifies the number of the dialer access group to which the specific interface belongs.
Step 8	**exit** **Example:** `Router(config-if)# exit`	Enters the global configuration mode.
Step 9	dialer-list dialer-group protocol protocol-name {permit | deny | list *access-list-number* | access-group} **Example:** `Router(config)# dialer-list 1 protocol ip list 1`	Creates a dialer list for traffic of interest and permits access to an entire protocol.

	Command or Action	Purpose
Step 10	access-list access-list-number permit *ip*-source-address **Example:** `Router(config)# access-list 1 permit any`	Defines traffic of interest.

Enabling 4G GPS and NMEA Data Streaming

GPS NMEA data streaming to external NMEA 2.0-compliant GPS plotter applications can be enabled on Cisco LTE.

Note For the LTE, the *unit* argument identifies the router slot, module slot, and the port, and is separated by slashes (0/2/0).

SUMMARY STEPS

1. configure terminal
2. controller cellular *unit*
3. lte gps enable
4. lte gps mode standalone
5. lte gps nmea {ip | udp [*source address*][*destination address*][*destination port*] }
6. test cellular *unit* modem-power-cycle
7. end
8. show cellular *unit* gps
9. show cellular *unit* gps detail

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Router# configure terminal`	Enters the configuration mode.
Step 2	controller cellular *unit* **Example:** `Router(config)# controller cellular 0/2/0`	Enters the controller cellular configuration mode.
Step 3	lte gps enable **Example:** `Router(config-controller)# lte gps enable`	(Optional) GPS is enabled by default. Use this command to enable the GPS feature if GPS has been disabled for any reason.
Step 4	lte gps mode standalone **Example:**	Enables the standalone GPS mode.

	Command or Action	Purpose
	`Router(config-controller)# lte gps mode standalone`	

Step 5	lte gps nmea {ip | udp [*source address*][*destination address*][*destination port*] }
	Example:
	`Router(config-controller)# lte gps nmea ip`
	or
	`Router(config-controller)# lte gps nmea`
		Enables NMEA. Cisco 4G LTE Advanced support only IP NMEA. Therefore, the IP interface and serial interface options are unavailable.

Step 6	test cellular *unit* modem-power-cycle **Example:** `Router# test cellular 0/2/0 modem-power-cycle`	GPS can take effect only after modem power cycle.
Step 7	end **Example:** `Router(config-controller)# end`	Exits the controller configuration mode and returns to the privileged EXEC mode.

Step 8	show cellular *unit* gps
	Example:
	`Router# show cellular 0/2/0 gps`
	
	`GPS Info`
	`-------------`
	`GPS Feature: enabled`
	`GPS Mode Configured: standalone`
	`GPS Port Selected: Dedicated GPS port`
	`GPS Status: GPS coordinates acquired`
	`Last Location Fix Error: Offline [0x0]`
	`Latitude: 38 Deg 11 Min 22.1939 Sec North`
	`Longitude: 96 Deg 40 Min 48.7066 Sec West`
	`Timestamp (GMT): Thu Jun 29 07:13:42 2017`
	
	`Fix type index: 0, Height: 318 m`
	`Satellite Info`
	`----------------`
	`Satellite #3, elevation 62, azimuth 282, SNR 53`
	`.`
	`.`
	`.`
	`Satellite #28, elevation 0, azimuth 0, SNR 0`
	`Router#`
		Displays a summary of the following GPS data:
		• GPS state information (GPS disabled, GPS acquiring, GPS enabled)
		• GPS mode configured (standalone)
		• GPS location and timestamp information
		• GPS satellite information
		• GPS feature (enabled or disabled)
		• GPS port selected (Dedicated GPS and GPS port with voltage-no-bias)

Step 9	show cellular *unit* gps detail
	Example:
	`Router# show cellular 0 gps detail`
	`GPS Info`
	`-------------`
	`GPS Feature: enabled`
	`GPS Mode Configured: standalone`
	`GPS Port Selected: Dedicated GPS port`
	`GPS Status: GPS coordinates acquired`
	`Last Location Fix Error: Offline [0x0]`
		Displays detailed GPS data.

	Command or Action	Purpose

	`Latitude: 38 Deg 11 Min 22.1939 Sec North`
	`Longitude: 96 Deg 40 Min 48.7066 Sec West`
	`Timestamp (GMT): Thu Jun 29 07:13:42 2017`
	`Fix type index: 0, Height: 0 m`
	`HDOP: , GPS Mode Used: not configured`
	
	`Satellite Info`
	`----------------`
	`Satellite #3, elevation 0, azimuth 0, SNR 53`
	`.`
	`.`
	`.`
	`Satellite #9, elevation 0, azimuth 0, SNR 0`
	`Router#`	

Configuring 4G SMS Messaging

Note For the LTE, the *unit* argument identifies the router slot, module slot, and the port, and is separated by slashes (0/2/0).

SUMMARY STEPS

1. configure terminal
2. controller cellular *unit*
3. lte sms archive path *FTP-URL*
4. cellular *unit* lte sms view { all | *ID* | summary }
5. end
6. show cellular *unit* sms
7. cellular *unit* lte sms send *number*
8. cellular *unit* lte sms delete [all | *id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal **Example:** `Router# configure terminal`	Enters the configuration mode.
Step 2	controller cellular *unit* **Example:** `Router(config)# controller cellular 0/2/0`	Enters the controller cellular configuration mode.
Step 3	lte sms archive path *FTP-URL* **Example:** `Router(config-controller)# lte sms archive path ftp://username:password@172.25.211.175/SMS-LTE`	Specifies an FTP server folder path to send all the incoming and outgoing SMS messages. After the folder path is identified, it is appended automatically with outbox and

	Command or Action	Purpose

		inbox folders for the path to which SMS messages are sent and received, for example:
		`ftp://172.25.211.175/SMS-LTE/outbox`
		`ftp://172.25.211.175/SMS-LTE/inbox`

Step 4	cellular *unit* lte sms view { all | *ID* | summary }
	Example:
	`Router# cellular 0/2/0 lte sms view summary`
	
	`ID FROM YY/MM/DD HR:MN:SC SIZE CONTENT`
	`0 4442235525 12/05/29 10:50:13 137 Your entry last month has...`
	`2 5553337777 13/08/01 10:24:56 5 First`
	`3 5553337777 13/08/01 10:25:02 6 Second`
		Displays the message contents of incoming texts received by a modem.
		• **all**—Displays the message contents of up to 255 incoming text messages received by the modem.
		• *ID*—Displays the message contents for a specified ID (0-255) of an incoming text message.
		• **summary**—Displays a summary of the incoming text messages received by the modem.

Step 5	end **Example:** `Router# end`	Exits the configuration mode and returns to the privileged EXEC mode.

Step 6	show cellular *unit* sms
	Example:
	`Router#  show cellular 0/2/0 sms`
	`Incoming Message Information`
	`----------------------------`
	`SMS stored in modem = 20`
	`SMS archived since booting up = 0`
	`Total SMS deleted since booting up = 0`
	`Storage records allocated = 25`
	`Storage records used = 20`
	`Number of callbacks triggered by SMS = 0`
	`Number of successful archive since booting up = 0`
	`Number of failed archive since booting up = 0`
	
	`Outgoing Message Information`
	`----------------------------`
	`Total SMS sent successfully = 0`
	`Total SMS send failure = 0`
	`Number of outgoing SMS pending = 0`
	`Number of successful archive since booting up = 0`
	`Number of failed archive since booting up = 0`
	`Last Outgoing SMS Status = SUCCESS`
	`Copy-to-SIM Status = 0x0`
	`Send-to-Network Status = 0x0`
	`Report-Outgoing-Message-Number:`
	`Reference Number = 0`
	`Result Code = 0x0`
	`Diag Code = 0x0 0x0 0x0 0x0 0x0`
	
	`SMS Archive URL = ftp://lab:lab@1.3.150.1/outbox`
		Displays all the information in the text messages sent and received. Message information includes text messages sent successfully, received, archived, and messages pending to be sent. LTE-specific information on errors in case of a FAILED attempt may also be displayed.

Step 7	cellular *unit* lte sms send *number* **Example:**	Enables a user to send a LTE band SMS message to other valid recipients, provided they have a text message plan.

	Command or Action	Purpose
	`Router# cellular 0/2/0 lte sms send 15554443333 <sms text>`	The *number* argument is the telephone number of the SMS message recipient. **Note** 10-digit or 11-digit (phone) numbers are the proper numerical format for sending a text. For example, ########## or 1##########. Seven digits are not supported.
Step 8	cellular *unit* lte sms delete [all | *id*] **Example:** `Router# cellular 0/2/0 lte sms delete [ all | id ]`	(Optional) Deletes one message ID or all of the stored messages from memory.

Configuring Modem DM Log Collection

Diagnostic Monitor (DM) Log is a modem's feature that captures data transactions between the modem and the network over the radio frequency interface. This feature is a useful tool for troubleshooting 3G and 4G data connectivity or performance issues.

Once a DM log file is captured, diagnostic software tools, such as Sierra Wireless SwiLog and Qualcomm QXDM, can be used to decode the DM log file to understand the issues. A member of Cisco TAC can help with decoding the DM log files.

To configure DM log collection, enter the following commands, starting in privileged EXEC mode.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**controller cellular** *slot* **Example:** `Router(config)# controller cellular 0/2/0`	Enters cellular controller configuration mode.
Step 3	**lte modem dm-log** {**autoshop** {**link-down** | **timer** *time*} | **enable** | **filesize** *size* | **filter**} **bootflash:***file* | **flash:***file*} **rotation** | **size** *log-size*} **Example:** `Router(config-controller)# lte modem dm-log enable`	Configures DM logging for LTE modem. • **autostop**—Automatically stops DM log capturing based on: **link-down**—cellular interface link down event **timer***timer*—amount of time in minutes • **enable**—Starts DM log capturing.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td></td><td>

- **filesize** *size*—Specifies the maximum log file size, in MB for each DM log file before creating another DM log file. Range is from 1 to 64. Default is 20.
- **filter** *location***:***filename*—Specifies the DM log filter to use from the following locations:

  —bootflash:*file*

  —flash:*file*

  Note Bootflash and flash are the only valid locations to store the DM log filter file.

  Note If the DM log filter file is not specified, the generic filter file, which comes with the router will be used.

  Note The DM log filter file needs to be in .sqf format.
- **rotation**—Enables continuous DM log capturing by replacing the oldest DM log files with the latest.
- **size** *log-size*—Specifies the maximum total size in MB of all DM log files that can be allowed in the bootflash or flash before modem stops capturing DM log files. If rotation is enabled, the oldest DM files is replaced with the latest DM file to meet this size configuration.

</td></tr>
<tr><td>Step 4</td><td>

end

Example:

```
Router(config-controller)# end
```

</td><td>Returns to privileged EXEC mode.</td></tr>
<tr><td>Step 5</td><td>

show cellular *unit* **logs dm-log**

Example:

```
Router# show cellular 0/2/0 logs dm-log
Integrated DM logging is on
output path = Utility Flash
filter = MC74xx generic -
v11026_Generic_GSM_WCDMA_LTE_IP-no-data-packets.sqf
maximum log size = 0
maximum file size = 0
log rotation = disabled

33 packets sent to the modem, 4663 bytes, 0 errors
28521 packets received from the modem, 13500758
bytes, 0 input drops
28521 packets stored in utility flash, 13500758
bytes
```

</td><td>(Optional) Displays DM log configuration and statistics.</td></tr>
</table>

	Command or Action	Purpose
	current file size = 13500758 current log size = 13500758 total log size = 13500758 Utility Flash DM log files = (1) files	

Example

The following example shows how to:

- Specifies the maximum size of all DM log files that can be stored in bootflash or flash to 512 MB
- Specifies the maximum size of each DM log file to 32 MB
- Uses MC7xxx_GPS_Log.sqf DM log filter in the flash
- Enable rotation
- Enables DM log capturing

```
Router(config-controller)# controller cell 0/2/0
Router(config-controller)# lte modem dm-log filesize 512
```

```
Router(config-controller)# controller cell 0/2/0
Router(config-controller)# lte modem dm-log filesize 32
```

The following example shows how to specify the filter file for LTE:

```
Router(config-controller)# controller cell 0/2/0
Router(config-controller)# lte modem dm-log filter flash:MC7xxx_GPS_Log.sqf
```

The following example shows how to enable DM log rotation for LTE:

```
Router(config-controller)# controller cell 0/2/0
Router(config-controller)# lte modem dm-log rotation
```

The following example shows how to specify the maximum log size for LTE:

```
Router(config-controller)# controller cell 0/2/0
Router(config-controller)# lte modem dm-log enable
```

The following example shows how to enable DM log rotation for LTE:

```
Router(config-controller)# controller cell 0/2/0
Router(config-controller)# end
```

The following example shows how to specify the maximum log size for LTE:

```
Router(config-controller)# controller cell 0/2/0
Router(config-controller)# lte modem dm-log size 1024
```

The following example shows how to enable DM log rotation for LTE:

```
Router(config-controller)# controller cell 0/2/0
Router(config-controller)# end
```

The following example shows what was configured on the router for DM log feature:

```
Router#show running-config | section controller
controller Cellular 0/2/0
 lte modem dm-log filter flash:MC7xxx_GPS_Log.sqf
 lte modem dm-log size 512
 lte modem dm-log filesize 32
 lte modem dm-log rotation
```

```
 lte modem dm-log enable
 lte modem dm-log size 1024
```

The following displays DM log configuration and statistics

```
Router#show cellular 0/2/0 logs dm-log
Integrated DM logging is on
output path = Utility Flash
filter = flash:MC7xxx_GPS_Log.sqf
maximum log size = 536870912
maximum file size = 33554432
log rotation = enabled

32 packets sent to the modem, 3879 bytes, 0 errors
158324 packets received from the modem, 75971279 bytes, 0 input drops
158324 packets stored in utility flash, 75971279 bytes

current file size = 8863042
current log size = 75971279
total log size = 75971279
Utility Flash DM log files =  (3) files
end
```

The following shows the DM log files created:

```
Router#dir flash:dmlog*
Directory of bootflash:/dmlog*

Directory of bootflash:/

   27  -rw-    33554069   Jun 7 2018 18:08:46 -08:00  dmlog-slot2-20180607-180628.bin
   28  -rw-    33554168   Jun 7 2018 18:11:25 -08:00  dmlog-slot2-20180607-180846.bin
   29  -rw-    14188544   Jun 7 2018 18:12:37 -08:00  dmlog-slot2-20180607-181125.bin
2885718016 bytes total (521891840 bytes free)
lte modem dm-log size 1024
```

The following shows hot to disable/stop DM log capturing:

```
Router(config)#controller cellular 0/2/0
Router(config-controller)#no lte modem dm-log enable
Router(config-controller)#end
```

Enabling Modem Crashdump Collection

Modem crashdump collection is useful in debugging firmware crash. To collect crash data, the modem has to be pre-configured so that it will stay in memdump mode after a crash. Memdump mode is a special boot-and-hold mode for the memdump utility to collect crash data.

For earlier releases, the crashdump collection required the PC to be connected to the router using a USB cable or a special RJ45-USB cable on a non-HSPA+7 3G module.

As part of the 3G and 4G serviceability enhancement, the crashdump collection utility is integrated into Cisco IOS.

To enable modem crashdump collection, perform the following steps.

Note The integrated modem crashdump collection feature is supported only on 3G HSPA and LTE based SKUs.

Before you begin

Ensure that the following prerequisites are met before attempting to enable crashdump logging:

- The modem needs to be provisioned for modem crashdump collection. Contact Cisco TAC for details.
- The modem should be in crash state. Run tests that will result in modem firmware crash. A “MODEM_DOWN” message on the router console or syslog is indicative of modem firmware crash.

Note After the modem firmware crashes, the modem is available for crashdump log collection only. Data calls cannot be made.

Procedure

	Command or Action	Purpose
Step 1	test { cell-cwan } *unit* modem-crashdump { on *location* | off } **Example:** `Router# test cell-host 0/2/0 modem-crashdump on local_uf`	Enables or disables modem crashdump collection. • **cell-host** —Keyword for fixed platform. • **cell-cwan** — Keyword for LTE on a modular inside platform. • *unit* —For LTE module, this is the router slot, module slot, and port separated by slashes (for example, 0/2/0). For fixed platform, this is the number 0. • **on** Enables crashdump log collection. • *location* —Specifies the destination URL where the modem crashdump logs will be stored. • **off** —Disables crashdump log collection.

Displaying Modem Log Error and Dump Information

As part of the 3G serviceability enhancement, commands strings (**at!err** and **at!gcdump**) can be sent to the modem using Cisco IOS CLI rather than setting up a reverse telnet session to the cellular modem to obtain log error and dump information.

To obtain log error and dump information, perform the following steps.

Note The modem log error and dump collection feature is supported only on 3G SKUs.

Procedure

	Command or Action	Purpose
Step 1	**show cellular** *unit* **log error** **Example:** `Router# show cellular 0/2/0 log error`	Shows modem log error and dump information.
Step 2	**test cellular** *unit* **modem-error-clear** **Example:** `Router# test cellular 0/2/0 modem-error-clear`	(Optional) Clears out the error and dump registers. By default, error and dump registers are not cleared out after a read. This command changes the operation so that registers are cleared once they are read. As a result, the AT command strings are changed to "**at!errclr=–1**" for CDMA and "**at!err=0**" for GSM modems.

Verifying the LTE Router Information

You can verify the configuration by using the following show commands:

show version

```
Router#show version
Cisco IOS XE Software, Version BLD_V166_THROTTLE_LATEST_20170622_080605_V16_6_0_237
Cisco IOS Software [Everest], ISR Software (ARMV8EB_LINUX_IOSD-UNIVERSALK9_IAS-M),
Experimental Version 16.6.20170622:072729
[v166_throttle-/scratch/mcpre/BLD-BLD_V166_THROTTLE_LATEST_20170622_080605 108]
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Thu 22-Jun-17 03:39 by mcpre


Cisco IOS-XE software, Copyright (c) 2005-2017 by cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

Router uptime is 2 hours, 16 minutes
Uptime for this control processor is 2 hours, 18 minutes
System returned to ROM by Reload Command
System image file is
"bootflash:c1100-universalk9_ias.BLD_V166_THROTTLE_LATEST_20170622_080605_V16_6_0_237.SSA.bin"
Last reload reason: Reload Command
```

```
This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.



Suite License Information for Module:'esg'

--------------------------------------------------------------------------------
Suite                 Suite Current         Type           Suite Next reboot
--------------------------------------------------------------------------------

Technology Package License Information:

-----------------------------------------------------------------
Technology    Technology-package           Technology-package
              Current       Type           Next reboot
------------------------------------------------------------------

cisco C1111-8PLTEAW (1RU) processor with 1464691K/6147K bytes of memory.
Processor board ID FGL21071SK4
1 Virtual Ethernet interface
11 Gigabit Ethernet interfaces
2 Cellular interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
6598655K bytes of flash memory at bootflash:.
978928K bytes of USB flash at usb0:.
0K bytes of WebUI ODM Files at webui:.
```

show platform

```
router# show platform
Chassis type: C1111-8PLTELAWN

Slot      Type                State                 Insert time (ago)
--------- ------------------- --------------------- -----------------
0         C1111-8PLTELAWN     ok                    00:04:56
 0/0      C1111-2x1GE         ok                    00:02:41
 0/1      C1111-ES-8          ok                    00:02:40
 0/2      C1111-LTE           ok                    00:02:41
 0/3      ISR-AP1100AC-N      ok                    00:02:41
R0        C1111-8PLTELAWN     ok, active            00:04:56
F0        C1111-8PLTELAWN     ok, active            00:04:56
P0        PWR-12V             ok                    00:04:30

Slot      CPLD Version        Firmware Version
--------- ------------------- ---------------------------------------
0         17100501            16.6(1r)RC3
```

```
R0        17100501            16.6(1r)RC3
F0        17100501            16.6(1r)RC3
```

show interfaces

```
router#sh interface cellular 0/2/0
Cellular0/2/0 is up, line protocol is up
  Hardware is LTE Adv CAT6 - Europe/North America Multimode LTE/DC-HSPA+/HSPA+/HSPA/UMTS/
  Internet address is 192.0.2.1/32
  MTU 1500 bytes, BW 50000 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive not supported
  DTR is pulsed for 1 seconds on reset
  Last input never, output 00:00:42, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     5 packets input, 460 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     21 packets output, 1692 bytes, 0 underruns
     0 output errors, 0 collisions, 8 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
router#
```

Configuring Cellular Modem Link Recovery

The cellular modem link recovery feature is disabled by default. It is recommended to enable the link recovery feature for improved performance and reliability.

When enabled, the feature monitors specific parameters such as RSSI (Received Signal Strength Indicator), RSRP (Reference Signal Received Power), and RSRQ (Reference Signal Received Quality), one at a time.

These parameters provide information about the strength and quality of the cellular signal.

The modem link recovery feature triggers the modem to reload when any of the configured values (RSSI, RSRP or RSRQ) go beyond the set threshold. Modem link recovery essentially restarts the cellular modem to re-establish a stable connection.

Note This feature does not automatically select the next best carrier network or initiate a SIM switchover based on the RSSI, RSRQ, RSRP values. It only focuses on reloading the modem to resolve potential connectivity problems.

To configure and enable the monitoring parameters for link recovery, perform the **lte modem link-recovery rssi onset-threshold** command for RSSI, **lte modem link-recovery rsrp onset-threshold** for RSRP and **lte modem link-recovery rsrq onset-threshold** for RSRQ.

To disable the link recovery feature, use:

{ lte } modem link-recovery disable | no lte | modem link-recovery disable }

> **Note** The link-recovery feature enables the RSRP (Reference Signal Received Power) and RSRQ (Reference Signal Received Quality) parameters on cellular modems from Cisco IOS XE Dublin 17.11.1a onwards.

To enable or disable the cellular modem link recovery feature (if required) perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **controller cellular** *unit*
3. For LTE modems, RSSI, RSRP (Reference Signal Received Power) and RSRQ (Reference Signal Received Quality) are recommended indicators of signal quality. Perform the **lte modem link-recovery rssi onset-threshold** command for RSSI, **lte modem link-recovery rsrp onset-threshold** for RSRP and **lte modem link-recovery rsrq onset-threshold** for RSRQ. To disable the link recovery feature, use: {**lte**} **modem link-recovery disable | no lte | modem link-recoverydisable**}
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**controller cellular** *unit* **Example:** `Router(config)# controller cellular 0/2/0`	Enters cellular controller configuration mode.

Step 3	For LTE modems, RSSI, RSRP (Reference Signal Received Power) and RSRQ (Reference Signal Received Quality) are recommended indicators of signal quality. Perform the **lte modem link-recovery rssi onset-threshold** command for RSSI, **lte modem link-recovery rsrp onset-threshold** for RSRP and **lte modem link-recovery rsrq onset-threshold** for RSRQ. To disable the link recovery feature, use: {**lte**} **modem link-recovery disable | no lte | modem link-recoverydisable**}
	
	Example:
	
	`Router(config-controller)# lte modem link-recovery disable`
	`Router(config-controller)# no lte modem link-recovery disable`
	
	`Router#show run | sec controller Cellular 0/2/0`
	`controller Cellular 0/2/0`
		Enables or disables the cellular modem link recovery feature (the cellular modem link recovery feature is disabled by default).
		
		Further enables the RSSI, RSRQ and RSRP parameters recommended for the link-recovery feature.
		
		Once we enable link-recovery, the default Cisco recommended values for link-recovery parameters are populated.
		
		We can change the values of link recovery parameters from the default Cisco recommended values, by using CLI for each parameter like in example.
		
		Note Changing the default recommended Cisco values is not advised as it will impact ideal performance of linkrecovery feature.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td>

```
lte modem link-recovery rssi
onset-threshold -110
lte modem link-recovery monitor-timer 20
lte modem link-recovery wait-timer 10
lte modem link-recovery debounce-count
6
```

For the RSSI parameter:

```
Router#configure terminal
Router(config)#controller Cellular 0/2/0
Router(config-controller)#lte modem
link-recovery
monitor-timer 30
Router(config-controller)#lte modem
link-recovery wait-timer 15
Router(config-controller)#lte modem
link-recovery debounce-count 8
Router(config-controller)#lte modem link-recovery
 rssi
onset-threshold -100
```

For the RSRQ parameter:

```
Router#configure terminal
Router(config)#controller
Cellular 0/2/0
Router(config-controller)#lte
modem rsrq onset-threshold -
19
```

For the RSRP parameter:

```
Router#configure terminal
Router(config)#controller
Cellular 0/2/0
Router(config-controller)#lte
modem rsrp onset-threshold -
139
```

</td><td>

Note Only one of the three parameters (RSSI, RSRP, RSRQ) can be configured at a time. If no parameter is explicitly set by the user when link recovery is enabled, the system will fall back to the default value of RSSI.

</td></tr>
<tr><td>Step 4</td><td>

end

Example:

```
Router(config)# end
```

</td><td>Exits the configuration mode and returns to the privileged EXEC mode.</td></tr>
</table>

Cellular Modem Link Recovery Parameters

There are three configurable parameters to adjust the behavior of cellular link recovery. The default values optimized for the best performance of the feature and changing it is not recommended unless advised by Cisco.

The following table explains the link recovery parameters.:

Table 50: Link Recovery Parameters

Parameter	Description
rssi onset-threshold	This parameter defines the RSSI value below which the link recovery feature triggers additional scrutiny to look for potential issues and take action if needed. The range of this parameter can be set from -90 dBm to -125 dBm. The recommended and default value is -110 dBm.
monitor-timer	This parameter determines how often link recovery looks for potential issues. The default value for this parameter is 20 seconds meaning that link recovery feature will be triggered every 20 seconds and look at certain parameters to determine if there is a potential issue. You can configure the monitor-timer range between 20 to 60 seconds. Increasing the monitor timer value above 20 seconds will increase the response time of the feature.
wait-timer and debounce-count	The wait-timer parameter is used in conjunction with the debounce-count parameter to perform more frequent, additional checks, once the link recovery feature has identified a potential issue that needs to be recovered from, with a modem power-cycle. The default value for wait-timer is 10 seconds and the default value for debounce- count is 6. With this setting, once link recovery has identified an inoperative modem state, it performs additional checks every 10 seconds, up to 6 times, to determine if the issue has been resolved without a modem power-cycle. Reducing the debounce-count and the wait-timer makes faster link recovery, while reducing them may increase the time for recovery. The configurable range for wait-timer is 5-60 seconds. The configurable range for debounce-count is 6-20 seconds.

Verifying the Cellular Modem Link Recovery Configuration

To determine if the cellular modem link recovery is enabled, use the **show controller cellularunit** command. In this example, the cellular modem link recovery feature related information is highlighted.

```
Router# show controller cellular 0/2/0Interface Cellular0/2/0
LTE Module - Multimode LTE/DC-HSPA+/HSPA+/HSPA/UMTS/EDGE/GPRS unit 2

Cellular Modem Configuration
==============================
Modem is recognized as valid
Power save mode is OFF
manufacture id =  0x00001199     product id = 0x000068C0
Sierra Wireless unknown modem
Modem Uplink Speed = 50000 kbit.
```

```
Modem Downlink Speed = 300000 kbit.

GPS Feature = enabled
GPS Status =  NMEA Disabled
GPS Mode = not configured

Cellular Dual SIM details:
---------------------------
SIM 0 is present
SIM 1 is not present
SIM 0 is active SIM

Module Reload Statistics
-------------------------
Soft OIR reloads = 0
Hard OIR reloads = 0
-------------------------

Modem Management Statistics
---------------------------
Modem resets = 1
Modem timeouts = 0
Link recovery is ON

Registration check is ON
RSSI threshold value is -110 dBm
Monitor Timer value is 20 seconds
Wait Timer value is 10 seconds
Debounce Count value is 6

Link recovery count is 0
```

When the cellular modem link recovery occurs and modem is power cycled, you can see the %CELLWAN-2-MODEM_DOWN message on the console logs and additionally there is a %CELLWAN-2-LINK_RECOVERY message which indicates that action has been taken by the cellular modem link recovery feature.

Whenever the cellular modem link recovery has occurred, it updates the Modem timeouts counter under the Modem Management Statistics section of the show controller cellular unit command output. Modem parameters at the last timeout section has information that helps to identify the cause of the issue that triggered link recovery

In the following example log, the messages, modem time out counter, and modem parameters at the last time out are highlighted.

***Jul 19 17:15:18.980 PDT: %CELLWAN-2-LINK_RECOVERY: Cellular0/1/0: Cellular Modem has been power cycled**

```
Device#show controller Cellular 0/2/0
Interface Cellular0/2/0
LTE Module - Multimode LTE/DC-HSPA+/HSPA+/HSPA/UMTS/EDGE/GPRS unit 2

Cellular Modem Configuration
==============================
Modem is recognized as valid
Power save mode is OFF
manufacture id =  0x00001199     product id = 0x000068C0
Sierra Wireless unknown modem
Modem Uplink Speed = 50000 kbit.
Modem Downlink Speed = 300000 kbit.

GPS Feature = enabled
GPS Status =  NMEA Disabled
```

```
GPS Mode = not configured

Cellular Dual SIM details:
---------------------------
SIM 0 is present
SIM 1 is not present
SIM 0 is active SIM

Module Reload Statistics
-------------------------
Soft OIR reloads = 0
Hard OIR reloads = 0
-------------------------
 Modem Management Statistics
---------------------------
Modem resets = 1
Modem user initiated resets = 0
Modem user initiated power-cycles = 0
Modem timeouts = 1
Modem parameters at the last timeout:
         LTE first time attach State was No
         Radio Interface Technology Mode was AUTO
         Operating Mode was Online
         RSSI was -0 dBm
         Packet switch domain status was Not Attached
         Registration state(EMM) was Not Registered
         Downlink traffic was not present
Link recovery is ON
Registration check is ON
RSSI threshold value is -110 dBm
Monitor Timer value is 20 seconds
Wait Timer value is 10 seconds
Debounce Count value is 6
```

Configuration Examples for 3G and 4G Serviceability Enhancement

Example: Sample Output for the show cellular logs dm-log Command

The following shows a sample output of the **show cellular logs dm-log** command:

```
Router# show cellular 0/2/0 logs dm-log
Integrated DM logging is on
filter = generic
maximum log size = 67108864
maximum file size = 20971520
log rotation = disabled
7 packets sent to the modem, 3232 bytes, 0 errors
75 packets received from the modem, 57123 bytes, 0 input drops
75 packets stored in file system, 57123 bytes, 0 errors, 0 aborts
2 max rcv queue size
current file size = 57123
current log size = 57123
total log size = 57123
DM log files: (1 files)
```

Example: Sample Output for the show cellular logs modem-crashdump Command

The following shows a sample output of the **show cellular logs modem-crashdump** command:

```
Router# show cellular 0/2/0 logs modem-crashdump
Modem crashdump logging: off
Progress = 100%
Last known State = Getting memory chunks
Total consecutive NAKs = 0
Number of retries = 0
Memory Region Info:
1: Full SDRAM [Base:0x0, Length:0x2000000]
2: MDSP RAM A region [Base:0x91000000, Length:0x8000]
3: MDSP RAM B region [Base:0x91200000, Length:0x8000]
4: MDSP RAM C region [Base:0x91400000, Length:0xC000]
5: MDSP Register region [Base:0x91C00000, Length:0x28]
6: ADSP RAM A region [Base:0x70000000, Length:0x10000]
7: ADSP RAM B region [Base:0x70200000, Length:0x10000]
8: ADSP RAM C region [Base:0x70400000, Length:0xC000]
9: ADSP RAM I region [Base:0x70800000, Length:0x18000]
10: CMM Script [Base:0x6A350, Length:0x310]
Router#
```

Configuration Examples for LTE

Example: Basic Cellular Interface Configuration: Cisco LTE

The following example shows how to configure the cellular interface to be used as a primary and is configured as the default route:

```
Router# show running-config
interface Cellular 0/2/0
ip address negotiated
dialer in-band
dialer-group 1
ip route 172.22.1.10 255.255.255.255 cellular 0/2/0
dialer-list 1 protocol ip permit
```

Configuration Examples for Cisco LTE

The following example shows how to configure Cisco LTE:

```
Router# show running-config
Building configuration...
Current configuration : 2991 bytes
!
! Last configuration change at 21:31:48 UTC Mon May 18 2015
!
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
service internal
```

```
no platform punt-keepalive disable-kernel-core
platform shell
!
hostname C1111-LTEEA
!
boot-start-marker
!
!
!
logging buffered 10000000
no logging console
enable password lab
!
no aaa new-model
!
!
!
!
!
!
!
subscriber templating
!
multilink bundle-name authenticated
icense udi pid ISR4321/K9 sn FDO181701PZ
!
spanning-tree extend system-id
!
!
redundancy
 mode none
!
!
!
!
controller Cellular 0/2/0
 lte sim data-profile 16 attach-profile 16
 lte gps mode standalone
 lte gps nmea
 lte modem link-recovery disable

 interface GigabitEthernet0/0/1
 ip address 192.0.2.1 255.255.255.0
 ip nat outside

 negotiation auto
!
interface Cellular0/2/0
 ip address negotiated
 ip nat outside
 dialer in-band
 dialer idle-timeout 0
 dialer watch-group 1
 dialer-group 1
 pulse-time 1
!
interface Cellular0/2/1
 no ip address
 shutdown
 dialer in-band
 pulse-time 1
!
!
interface Vlan1
```

```
 no ip address
!
no ip nat service dns tcp
no ip nat service dns udp
ip nat inside source list 1 interface Cellular0/2/0 overload
ip forward-protocol nd
ip http server
no ip http secure-server
ip http max-connections 16
ip tftp source-interface GigabitEthernet0/0/1
ip dns server
ip route 192.0.2.2 192.0.2.3 Cellular0/2/0
ip route 223.255.254.0 255.255.255.0 1.3.0.1
!
!
access-list 1 permit 192.0.2.5 255.255.255.255
dialer watch-list 1 ip 192.0.2.6 255.255.255.255
dialer-list 1 protocol ip permit
!
snmp-server community public RO
snmp-server community private RW
snmp-server community lab RW
snmp-server host 192.0.2.1 public
snmp-server manager
control-plane
!
!
line con 0
 exec-timeout 0 0
 stopbits 1
line aux 0
 exec-timeout 0 0
 stopbits 1
line vty 0 4
 login
 transport input all
!
!
end
```

Cellular Back-off: Example

The following example shows how to configure the cellular back-off feature to stop continuous session activation requests back to the router:

```
Router#show cell 0/2/0 all
Profile 1, Packet Session Status = INACTIVE
Profile 2, Packet Session Status = INACTIVE
Profile 3, Packet Session Status = INACTIVE
.
.
.
Profile 16, Packet Session Status = INACTIVE
Router#
Router#show cell 0/2/0 c n
Current System Time = Sun Jan 6 0:8:37 1980
Current Service Status = Normal
Current Service = Packet switched
Current Roaming Status = Roaming
Network Selection Mode = Automatic
Network = 123 456
Mobile Country Code (MCC) = 123
Mobile Network Code (MNC) = 456
```

```
Packet switch domain(PS) state = Attached
LTE Carrier Aggregation state = Deconfigured
Registration state(EMM) = Registered
EMM Sub State = Normal Service
Tracking Area Code (TAC) = 1801
Cell ID = 768001
Network MTU is not Available
Router#
Router#ping 192.0.2.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.192.187.254, timeout is 2 seconds:

*Dec 20 23:22:28.025: %CELLWAN-6-CELLULAR_BACKOFF_START: Cellular0/2/0: Cellular back-off
has started on PDN 0....
Success rate is 0 percent (0/5)
Router#

Router#ping 192.0.2.2
Type escape sequence to abort.
RouterSending 5, 100-byte ICMP Echos to 192.0.2.2, timeout is 2 seconds
.
.
.
Router#show cell 0/2/0
Profile 1, Packet Session Status = INACTIVE
Profile 2, Packet Session Status = INACTIVE
Profile 3, Packet Session Status = INACTIVE
Router Call end mode = 3GPP
Router Session disconnect reason type = 3GPP specification defined(6)
Session disconnect reason = Option unsubscribed(33)
Enforcing cellular interface back-off
Period of back-off = 1 minute(s)
Profile 4, Packet Session Status = INACTIVE
...
Profile 16, Packet Session Status = INACTIVE
Router#
Router#show cell 0/2/0 cn
Sending 5, 100-byte ICMP Echos to 192.0.2.2, timeout is 2 seconds:
Router.....
Success rate is 0 percent (0/5)
Router#
Router#ping 192.0.2.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.0.2.5, timeout is 2 seconds:
Router.....
Success rate is 0 percent (0/5)
Router#show cell 0/2/0 cping 192.0.2.6  Type escape sequence to abort.
RouterSending 5, 100-byte ICMP Echos to 192.0.2.6 , timeout is 2 seconds:
Router.....
RouterSuccess rate is 0 percent (0/5)
Router#ping 192.0.2.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.0.2.6 , timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
Router#ping 192.0.2.6
Router#sh cell 0/2/0 c
Profile 1, Packet Session Status = INACTIVE
Profile 2, Packet Session Status = INACTIVE
Profile 3, Packet Session Status = INACTIVE
RouterCall end mode = 3GPP
RouterSession disconnect reason type = 3GPP specification defined(6)
RouterSession disconnect reason = Option unsubscribed(33)
RouterEnforcing cellular interface back-off
```

```
 Period of back-off = 1 minute(s)
Profile 4, Packet Session Status = INACTIVE
...
Profile 16, Packet Session Status = INACTIVE
Profile 4, Packet Session Status = INACTIVE
Profile 5, Packet Session Status = INACTIVE
.
.
.
Profile 16, Packet Session Status = INACTIVE
```

Example: GRE Tunnel over Cellular Interface Configuration

The following example shows how to configure the static IP address when a GRE tunnel interface is configured with **ip address unnumbered** *cellular interface*:

Note The GRE tunnel configuration is supported only if the service providers provide a public IP address on the LTE interface.

Note For service providers using a private IP address, the point-to-point static GRE tunnel cannot be set up with a private IP address at one end and a public IP address on the other end.

```
interface Tunnel2
ip unnumbered <internal LAN interface GE0/0 etc.>
tunnel source Cellular0/2/0
tunnel destination a.b.c.d
interface Cellular0/2/0
ip address negotiated
no ip mroute-cache
dialer in-band
dialer-group 1
```

Example: LTE as Backup with NAT and IPSec

The following example shows how to configure the LTE on the router as backup with NAT and IPsec:

The receive and transmit speeds cannot be configured. The actual throughput depends on the cellular network service.

For service providers using a private IP address, use the **crypto ipsec transform-set esp** command (that is, esp-aes esp-sha256-hmac...).

```
ip dhcp excluded-address 10.4.0.254
!
ip dhcp pool lan-pool
   network 10.4.0.0 255.255.0.0
   dns-server 10.4.0.254
   default-router 10.4.0.254
!
!
crypto isakmp policy 1
```

```
 encr 3des
 authentication pre-share
crypto isakmp key  address a.b.c.d
!
!
crypto ipsec transform-set  ah-sha-hmac esp-3des
!
crypto map gsm1 10 ipsec-isakmp
 set peer a.b.c.d
 set transform-set
 match address 103
!
interface ATM0/2/0
 no ip address
 ip virtual-reassembly
 load-interval 30
 no atm ilmi-keepalive
 dsl operating-mode auto
!
interface ATM0/2/0.1 point-to-point
 backup interface Cellular0/2/0
ip address negotiated
ip mtu 1492
ip nat outside
ip virtual-reassembly
encapsulation ppp
load-interval 30
dialer pool 2
dialer-group 2
ppp authentication chap callin
ppp chap hostname cisco@dsl.com
ppp chap password 0 cisco
ppp ipcp dns request
crypto map gsm1

 ip nat outside
 ip virtual-reassembly
 no snmp trap link-status
 pvc 0/35
  pppoe-client dial-pool-number 2
 !
!
interface Cellular0/2/0
 ip address negotiated
 ip nat outside
 ip virtual-reassembly
no ip mroute-cache
 dialer in-band
 dialer idle-timeout 0
dialer-group 1
 crypto map gsm1
!
interface Vlan1
 description used as default gateway address for DHCP clients
 ip address 10.4.0.254 255.255.0.0
 ip nat inside
 ip virtual-reassembly
!
ip local policy route-map track-primary-if
ip route 0.0.0.0 0.0.0.0 Dialer2 track 234
ip route 0.0.0.0 0.0.0.0 Cellular0/3/0 254
!
!
ip nat inside source route-map nat2cell interface Cellular0/2/0 overload
```

```
ip nat inside source route-map nat2dsl overload
!
ip sla 1
 icmp-echo 2.2.2.2 source
 timeout 1000
 frequency 2
ip sla schedule 1 life forever start-time now
access-list 1 permit any
access-list 101 deny ip 10.4.0.0 0.0.255.255 10.0.0.0 0.255.255.255
access-list 101 permit ip 10.4.0.0 0.0.255.255 any
access-list 102 permit icmp any host 2.2.2.2
access-list 103 permit ip 10.4.0.0 0.0.255.255 10.0.0.0 0.255.255.255
dialer-list 1 protocol ip list 1
dialer-list 2 protocol ip permit
!
!
route-map track-primary-if permit 10
 match ip address 102
!
route-map nat2dsl permit 10
 match ip address 101
!
route-map nat2cell permit 10
 match ip address 101
 match interface Cellular0/2/0
!
exec-timeout 0 0
login
 modem InOut
```

Example: SIM Configuration

Locking the SIM Card

The following example shows how to lock the SIM. The italicized text in this configuration example is used to indicate comments and are not be seen when a normal console output is viewed.

```
Router# sh cellular 0/2/0 security
Card Holder Verification (CHV1) = Disabled
SIM Status = OK
SIM User Operation Required = None
Number of CHV1 Retries remaining = 3
Router# !! SIM is in unlocked state.!
Router# cellular 0/2/0 lte sim lock 1111
!!!WARNING: SIM will be locked with pin=1111(4).
Do not enter new PIN to lock SIM. Enter PIN that the SIM is configured with.
Call will be disconnected!!!
Are you sure you want to proceed?[confirm]
Router#
Apr 26 19:35:28.339: %CELLWAN-2-MODEM_DOWN: Modem in NIM slot 0/2 is DOWN
Apr 26 19:35:59.967: %CELLWAN-2-MODEM_UP: Modem in NIM slot 0/2 is now UP
Router#
Router# sh cellular 0/2/0 security
Card Holder Verification (CHV1) = Enabled
SIM Status = Locked
SIM User Operation Required = Enter CHV1
Number of CHV1 Retries remaining = 3
Router# !! SIM is in locked state.!
```

Unlocking the SIM Card

The following example shows how to unlock the SIM. The italicized text throughout this configuration example is used to indicate comments and will not be seen when a normal console output is viewed.

```
Router# sh cellular 0/2/0 security
Card Holder Verification (CHV1) = Enabled
SIM Status = Locked
SIM User Operation Required = Enter CHV1
Number of CHV1 Retries remaining = 3
Router# !! SIM is in locked state.!
Router# cellular 0/2/0 lte sim unlock 1111
!!!WARNING: SIM will be unlocked with pin=1111(4).
Do not enter new PIN to unlock SIM. Enter PIN that the SIM is configured with.
Call will be disconnected!!!
Are you sure you want to proceed?[confirm]
Router#
Router# sh cellular 0/2/0 security
Card Holder Verification (CHV1) = Disabled
SIM Status = OK
SIM User Operation Required = None
Number of CHV1 Retries remaining = 3
Router# !! SIM is in unlocked state.!
```

Automatic SIM Authentication

The following example shows how to configure automatic SIM authentication. The italicized text throughout this configuration example is used to indicate comments and will not be seen when a normal console output is viewed.

```
Router# show cellular 0/2/0 security
Card Holder Verification (CHV1) = Disabled
SIM Status = OK
SIM User Operation Required = None
Number of CHV1 Retries remaining = 3
Router# !! SIM is in unlocked state.!Router# cellular 0/2/0 lte sim lock 1111
!!!WARNING: SIM will be locked with pin=1111(4).
Do not enter new PIN to lock SIM. Enter PIN that the SIM is configured with.
Call will be disconnected!!!
Are you sure you want to proceed?[confirm]
Router#
Apr 26 21:22:34.555: %CELLWAN-2-MODEM_DOWN: Modem in NIM slot 0/2 is DOWN
Apr 26 21:23:06.495: %CELLWAN-2-MODEM_UP: Modem in NIM slot 0/2 is now UP
Router#
Router# sh cellular 0/2/0 security
Card Holder Verification (CHV1) = Enabled
SIM Status = Locked
SIM User Operation Required = Enter CHV1
Number of CHV1 Retries remaining = 3
Router# !! SIM is in locked state. SIM needs to be in locked state for SIM authentication
to ! work.!Router#
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# controller cellular 0/2/0
Router(config-controller)# lte sim authenticate 0 1111
CHV1 configured and sent to modem for verification
Router(config-controller)# end
Router#
Apr 26 21:23:50.571: %SYS-5-CONFIG_I: Configured from console by console
Router#
Router# sh cellular 0/2/0 security
```

```
Card Holder Verification (CHV1) = Enabled
SIM Status = OK
SIM User Operation Required = None
Number of CHV1 Retries remaining = 3
Router#!! SIM is now in locked state but it can be used for connectivity since authentication
 is ! good. Authentication can be saved in the router configuration so that when you boot
up ! the router with the same locked SIM, connection can be established with the correct !
 Cisco IOS configuration.!
```

Changing the PIN Code

The following example shows how to change the assigned PIN code. The italicized text throughout this configuration example is used to indicate comments and will not be seen when a normal console output is viewed.

```
Router# sh cellular 0/2/0 security
Card Holder Verification (CHV1) = Disabled
SIM Status = OK
SIM User Operation Required = None
Number of CHV1 Retries remaining = 3
Router#!! SIM is in unlocked state.!Router#
Router# cellular 0/2/0 lte sim lock 1111
!!!WARNING: SIM will be locked with pin=1111(4).
Do not enter new PIN to lock SIM. Enter PIN that the SIM is configured with.
Call will be disconnected!!!
Are you sure you want to proceed?[confirm]
Router#
Apr 26 21:58:11.903: %CELLWAN-2-MODEM_DOWN: Modem in NIM slot 0/2 is DOWN
Apr 26 21:58:43.775: %CELLWAN-2-MODEM_UP: Modem in NIM slot 0/2 is now UP
Router#
Router# sh cellular 0/2/0 security
Card Holder Verification (CHV1) = Enabled
SIM Status = Locked
SIM User Operation Required = Enter CHV1
Number of CHV1 Retries remaining = 3
Router#!! SIM is in locked state. SIM needs to be in locked state to change its PIN.!Router#
Router# cellular 0/2/0 lte sim change-pin 1111 0000
!!!WARNING: SIM PIN will be changed from:1111(4) to:0000(4)
Call will be disconnected. If old PIN is entered incorrectly in 3 attempt(s), SIM will be
blocked!!!
Are you sure you want to proceed?[confirm]
Resetting modem, please wait...
CHV1 code change has been completed. Please enter the new PIN in controller configuration
for verfication
Router#
Apr 26 21:59:16.735: %CELLWAN-2-MODEM_DOWN: Modem in NIM slot 0/2 is DOWN
Apr 26 21:59:48.387: %CELLWAN-2-MODEM_UP: Modem in NIM slot 0/2 is now UP
Router#
Router#
Router# sh cellular 0/2/0 security
Card Holder Verification (CHV1) = Enabled
SIM Status = Locked
SIM User Operation Required = Enter CHV1
Number of CHV1 Retries remaining = 3
Router#!! SIM stays in locked state, as expected, but with new PIN.!Router# cellular 0/2/0
 lte sim unlock 0000
!!!WARNING: SIM will be unlocked with pin=0000(4).
Do not enter new PIN to unlock SIM. Enter PIN that the SIM is configured with.
Call will be disconnected!!!
Are you sure you want to proceed?[confirm]
Router#
Router# show cellular 0/2/0 security
```

```
Card Holder Verification (CHV1) = Disabled
SIM Status = OK
SIM User Operation Required = None
Number of CHV1 Retries remaining = 3
Router#!! Unlock with new PIN is successful. Hence, changing PIN was successful.!
```

Configuring an Encrypted PIN

The following example shows how to configure automatic SIM authentication using an encrypted PIN. The italicized text throughout this configuration example is used to indicate comments and will not be seen when a normal console output is viewed.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# service password-encryption
Router(config)# username SIM privilege 0 password 1111
Router(config)# do sh run | i SIM
username SIM privilege 0 password 7 055A575E70.!! Copy the encrypted level 7 PIN. Use this
 scrambled PIN in the SIM authentication ! command.!

Router(config)# controller cellular 0/2/0
Router(config-controller)# lte sim authenticate 7 055A575E70
CHV1 configured and sent to modem for verification
Router(config-controller)# exit
Router(config)# no username SIM
Router(config)# end
May 14 20:20:52.603: %SYS-5-CONFIG_I: Configured from console by console
```

Upgrading the Modem Firmware

The following table describes the Sierra Wireless modems that are supported on Cisco LTE. The firmware for the modem is upgradable using Cisco IOS commands. The firmware is a Crossword Express (cwe) file and can be downloaded from the wireless software download page on Cisco.com.

Note Firmware upgrade is supported on utility flash.

Use only Cisco certified firmware. Using a firmware version not certified by Cisco may impact the wireless service provider network adversely.

Caution Do not disconnect power or switch the router off during the firmware upgrade process. This may result in permanent modem failure.

Note Firmware downgrade is not supported.

Table 51: Modem SKUs

SKU	Modem	Firmware	Release
EHWIC-4G-LTE-A	MC7700	MC7700	Cisco 16.6.1 or Later

Upgrading the Modem Firmware Manually With CLI

SUMMARY STEPS

1. Go to the Cisco Wireless WAN software download website at: http://software.cisco.com/download/navigator.html
2. On the Cisco Wireless WAN software page, go to **Products** -> **Cisco Interfaces and Modules** -> **Cisco High-Speed WAN interface Cards** and select your product from the list of available cards.
3. Select and download the appropriate firmware.
4. **terminal monitor**
5. **microcode reload cellular** pa-bay slot **modem-provision** [**flash**:<firmware_directory_name>]
6. **show cellular** *0/2/0* **hardware**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Go to the Cisco Wireless WAN software download website at: http://software.cisco.com/download/navigator.html	Provides access to Cisco Wireless WAN software downloads page to select the firmware for Cisco LTE. **Note** This website is only available to registered Cisco.com users.
Step 2	On the Cisco Wireless WAN software page, go to **Products** -> **Cisco Interfaces and Modules** -> **Cisco High-Speed WAN interface Cards** and select your product from the list of available cards.	Select your product for firmware upgrade.
Step 3	Select and download the appropriate firmware.	Download the modem firmware file to flash memory on the router.
Step 4	**terminal monitor** **Example:** `Router# terminal monitor`	Enables the logging console in privileged EXEC mode.
Step 5	**microcode reload cellular** pa-bay slot **modem-provision** [**flash**:<firmware_directory_name>] **Example:** `Router# microcode reload cellular 0 2 modem-provision bootflash:/<firmware directory>`	Initiates the firmware upgrade process. • pa-bay—Use 0 for LTE. • slot—For LTE, slot number, 0 to 3, where the LTE is plugged in. • For remote download, you can transfer this using the wireless link from Cisco.com onto flash.

	Command or Action	Purpose

Step 6	**show cellular** *0/2/0* **hardware**
	Example:
	`Router# show cellular 0 hardware`
	`Modem Firmware built = 2016/06/30 10:54:05`
	`Hardware Version = 1.0`
	`Device Model ID: EM7455`
		Verifies the firmware upgrade process.

EM74xx Manual Modem Firmware Upgrade: Example

```
Router#  sh cellu 0/2/0 hardware
Modem Firmware Version = SWI9X30C_02.20.03.00
Modem Firmware built = 2016/06/30 10:54:05
Hardware Version = 1.0
Device Model ID: EM7455
International Mobile Subscriber Identity (IMSI) = <imsi>
International Mobile Equipment Identity (IMEI) = <imei>
Integrated Circuit Card ID (ICCID) = <iccid>
Mobile Subscriber Integrated Services
Digital Network-Number (MSISDN) =
Modem Status = Modem Online
Current Modem Temperature = 44 deg C
PRI SKU ID = 1102526, PRI version = 002.020_000, Carrier = AT&T
OEM PRI version = 006
Router#cd fw_22_vzw
Router#dir
Directory of bootflash:/fw_22_vzw/

227586  -rw-         64389490  Jun 30 2000 10:21:29 +00:00  74XX_02.20.03.22.cwe
227587  -rw-            16951  Jun 30 2000 10:22:10 +00:00
7455_02.20.03.22_Verizon_002.026_000.nvu

6816092160 bytes total (5965422592 bytes free)
Router#cd
Router#microcode reload cellular 0 2 modem-provision bootflash:/fw_22_vzw/
Reload microcode? [confirm]
Log status of firmware download in router flash?[confirm]
Firmware download status will be logged in bootflash:fwlogfile
Microcode Reload Process launched for cwan slot/bay =0/2; hw type=0x102download option = 0

Router#Success !! send FW Upgrade command to card

*****************************************************
 The interface will be Shut Down for Firmware Upgrade
 This will terminate any active data connections.
*****************************************************
**************************
 Modem will be upgraded!
 Upgrade process will take up to 15 minutes. During
 this time the modem will be unusable.
 Please do not remove power or reload the router during
 the upgrade process.
***************************
*Jul  6 10:19:34.701: %LINK-5-CHANGED: Interface Cellular0/2/0, changed state to
administratively down
*Jul  6 10:19:34.701: %LINK-5-CHANGED: Interface Cellular0/2/1, changed state to
administratively down
----------------------------
FIRMWARE INFO BEFORE UPGRADE:
```

```
Modem Device ID: EM7455     MODEM F/W Boot Version: SWI9X30C_02.20.03.00
Modem F/W App Version: SWI9X30C_02.20.03.00     Modem SKU ID: 1102526
Modem Package Identifier:      Modem Carrier String: 4
Modem PRI Ver: 000.006     Modem Carrier Name: ATT
Modem Carrier Revision: 002.020_000
----------------------------
FW_UPGRADE: Modem needs CWE, PRI
*Jul  6 10:19:57.978: %CELLWAN-2-MODEM_DOWN: Modem in NIM slot 0/2 is DOWN
FW_UPGRADE: Upgrade begin at Thu Jul  6 10:20:01 2000
FW_UPGRADE: Upgrade end at Thu Jul  6 10:21:14 2000
FW_UPGRADE: Firmware upgrade success.....
FW_UPGRADE: Waiting for modem to become online
----------------------------
FIRMWARE INFO AFTER UPGRADE:
Modem Device ID: EM7455     MODEM F/W Boot Version: SWI9X30C_02.20.03.22
Modem F/W App Version: SWI9X30C_02.20.03.22     Modem SKU ID: 1102526
Modem Package Identifier:      Modem Carrier String: 5
Modem PRI Ver: 000.006     Modem Carrier Name: VERIZON
Modem Carrier Revision: 002.026_000
----------------------------
F/W Upgrade: Firmware Upgrade has Completed Successfully
*Jul  6 10:21:55.275: %CELLWAN-2-MODEM_RADIO: Cellular0/2/0 Modem radio has been turned on
*Jul  6 10:21:57.276: %LINK-3-UPDOWN: Interface Cellular0/2/0, changed state to down
*Jul  6 10:21:57.277: %LINK-3-UPDOWN: Interface Cellular0/2/1, changed state to down
Router#
Router#  sh cellu 0/2/0 hardware
Modem Firmware Version = SWI9X30C_02.20.03.22
Modem Firmware built = 2016/10/11 16:03:14
Hardware Version = 1.0
Device Model ID: EM7455
International Mobile Subscriber Identity (IMSI) =<imsi>
International Mobile Equipment Identity (IMEI) = <imei>
Integrated Circuit Card ID (ICCID) = <iccid>
Mobile Subscriber Integrated Services
Digital Network-Number (MSISDN) = <msisdn>
Modem Status = Modem Online
Current Modem Temperature = 0 deg C
PRI SKU ID = 1102526, PRI version = 002.026_000, Carrier = Verizon
OEM PRI version = 006
```

Configuring dm-log to Utility Flash: Example

```
Router(config)#controller cellular 0/2/0
Router(config-controller)#lte modem dm-log enable
Router(config-controller)#
*May 8 17:57:09.905: %SYS-5-CONFIG_I: Configured from console by console
Router#
Router#sh cell 0/2/0 log dm-log
Integrated DM logging is on
output path = Utility Flash
filter = bootflash:v11026_Generic_GPS.sqf
maximum log size = 0
maximum file size = 0
log rotation = disabled

32 packets sent to the modem, 4021 bytes, 0 errors
23668 packets received from the modem, 11131720 bytes, 0 input drops
23668 packets stored in utility flash, 11131720 bytes

current file size = 11131720
current log size = 11131720
total log size = 11131720
Utility Flash DM log files: (1) files
```

SNMP MIBs

Note It is recommended that you configure SNMP V3 with authentication/privacy when implementing SNMP SET operation.

The following Simple Management Network Protocol (SNMP) MIBs are supported on Cisco LTE:

- IF-MIB
- ENTITY-MIB
- CISCO-WAN-3G-MIB
- CISCO-WAN-CELL-EXT-MIB

For the CISCO-WAN-3G-MIB, the following tables and sub-tables are supported for 3G and LTE technologies:

- ciscoWan3gMIB(661)
- ciscoWan3gMIBNotifs(0)
- ciscoWan3gMIBObjects(1)
- c3gWanCommonTable(1)
- c3gWanGsm(3)
- c3gGsmIdentityTable(1)
- c3gGsmNetworkTable(2)
- c3gGsmPdpProfile(3)
- c3gGsmPdpProfileTable(1)
- c3gGsmPacketSessionTable(2)
- c3gGsmRadio(4)
- c3gGsmRadioTable(1)
- c3gGsmSecurity(5)
- c3gGsmSecurityTable(1)

For the CISCO-WAN-CELL-EXT-MIB, the following tables and sub-tables are supported for LTE technology only:

- ciscoWanCellExtMIB(817)
- ciscoWanCellExtMIBNotifs(0)
- ciscoWanCellExtMIBObjects(1)
- ciscoWanCellExtLte(1)

- cwceLteRadio(1)
- cwceLteProfile(2)

You can download the MIBs from the Cisco MIB Locator at http://www.cisco.com/go/mibs.

SNMP LTE Configuration: Example

The following example describes how to configure 3G 4G MIB trap on the router:

```
controller Cellular 0/2/0
 lte event rssi onset mib-trap All-lte
 lte event rssi onset threshold -100
 lte event rssi abate mib-trap All-lte
 lte event rssi abate threshold -90
 lte event temperature onset mib-trap
 lte event temperature onset threshold 55
 lte event temperature abate mib-trap
 lte event temperature abate threshold 50
 lte event modem-state mib-trap all
 lte event service mib-trap
 lte event network mib-trap
 lte event connection-status mib-trap All-lte
 lte event rsrp onset mib-trap All-lte
 lte event rsrp onset threshold -85
 lte event rsrp abate mib-trap All-lte
 lte event rsrp abate threshold -80
 lte event rsrq onset mib-trap All-lte
 lte event rsrq onset threshold -8
 lte event rsrq abate mib-trap All-lte
 lte event rsrq abate threshold -6
```

The following example describes how to configure SNMP capability on the router:

```
snmp-server group neomobilityTeam v3 auth notify 3gView
snmp-server view 3gView ciscoWan3gMIB included
snmp-server community neomobility-test RW snmp-server community public RW
snmp-server enable traps c3g
snmp server enable traps LTE
snmp-server host 172.19.153.53 neomobility c3g snmp-server host 172.19.152.77 public c3g
snmp-server host 172.19.152.77 public udp-port 6059
```

The following example describes how to configure an external host device to communicate with the router through SNMP:

```
setenv SR_MGR_CONF_DIR /users/<userid>/mibtest
setenv SR_UTIL_COMMUNITY neomobility-test
setenv SR_UTIL_SNMP_VERSION -v2c
setenv SR_TRAP_TEST_PORT 6059
```

Troubleshooting

This section provides the essential information and resources available for troubleshooting the Cisco LTE Support feature.

Verifying Data Call Setup

To verify the data call setup, follow these steps:

1. After you create a modem data profile using the cellular profile create command and configuring DDR on the cellular interface, send a ping from the router to a host across the wireless network.
2. If the ping fails, debug the failure by using the following debug and show commands:
3. **debug chat**
4. **debug modem**
5. **debug dialer**
6. **show cellular all**
7. **show controller cell***0/2/0*
8. **show interface cellular**
9. **show running-config**
10. **show ip route**
11. **show platform**
12. Save the output from these commands and contact your system administrator.

Checking Signal Strength

If the Received Signal Strength Indication (RSSI) level is very low (for example, if it is less than –110 dBm), follow these steps:

SUMMARY STEPS

1. Check the antenna connection. Make sure the TNC connector is correctly threaded and tightened.
2. If you are using a remote antenna, move the antenna cradle and check if the RSSI has improved.
3. Contact your wireless service provider to verify if there is service availability in your area.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Check the antenna connection. Make sure the TNC connector is correctly threaded and tightened.	
Step 2	If you are using a remote antenna, move the antenna cradle and check if the RSSI has improved.	
Step 3	Contact your wireless service provider to verify if there is service availability in your area.	

Verifying Service Availability

The following is a sample output for the **show cellular all** command for a scenario where the antenna is disconnected and a modem data profile has not been created.

```
Router# show cellular 0/2/0 all
Hardware Information
====================
Modem Firmware Version = SWI9X30C_02.20.03.00
Modem Firmware built = 2016/06/30 10:54:05
Hardware Version = 1.0
Device Model ID: EM7455
International Mobile Subscriber Identity (IMSI) = 123456000031546
International Mobile Equipment Identity (IMEI) = 356129070052334
Integrated Circuit Card ID (ICCID) = 8949001508130031546
Mobile Subscriber Integrated Services
Digital Network-Number (MSISDN) =
Modem Status = Modem Online
Current Modem Temperature = 42 deg C
PRI SKU ID = 1102526, PRI version = 002.017_000, Carrier = Generic
OEM PRI version = 002

Profile Information
====================

Profile 1 = ACTIVE* **
--------
PDP Type = IPv4v6
PDP address = 29.29.29.196
PDP IPV6 address = 2001:2678:2680:5FD7:DDE7:70E1:DC07:CCB7/64  Scope: Global
Access Point Name (APN) = broadband
Authentication = None
        Primary DNS address = 8.0.0.8
        Secondary DNS address = 8.8.4.4
        Primary DNS IPV6 address = 2001:4860:4860:0:0:0:0:8888
        Secondary DNS IPV6 address = 2001:4860:4860:0:0:0:0:8844

Profile 2 = ACTIVE
--------
PDP Type = IPv4v6
PDP address = 21.21.21.206
PDP IPV6 address = 2001:567A:567A:1480:5DD6:18D1:BD63:49DA/64  Scope: Global
Access Point Name (APN) = basic
Authentication = None
        Primary DNS address = 171.70.168.183
        Secondary DNS address = 8.8.8.8
        Primary DNS IPV6 address = 2001:4860:4860:0:0:0:0:8888
        Secondary DNS IPV6 address = 2001:4860:4860:0:0:0:0:8844

Profile 3 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = mpdn
Authentication = None

Profile 4 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = broadband
Authentication = None

Profile 5 = INACTIVE
```

```
--------
PDP Type = IPv4
Access Point Name (APN) = cisco.gw4.vzwentp
Authentication = None

Profile 6 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = mobility-de1
Authentication = None

Profile 7 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = mobility-de2
Authentication = None

Profile 8 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = broadband
Authentication = None

Profile 9 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = mpdndt-qos
Authentication = None

Profile 10 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = mobility-de2
Authentication = None

Profile 11 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = broadband
Authentication = None

Profile 12 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = wfqos
Authentication = CHAP
Username: ipv4v6
Password:

Profile 13 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = broadband
Authentication = CHAP
Username: ipv4v6
Password:

Profile 14 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = mobility-de2
Authentication = CHAP
Username: ipv4v6
Password:
```

```
Profile 15 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = aaaauth
Authentication = CHAP
Username: ipv4v6
Password:

Profile 16 = INACTIVE
--------
PDP Type = IPv4
Access Point Name (APN) = broadband
Authentication = CHAP
Username: ipv4v6
Password:

  * - Default profile
 ** - LTE attach profile


Configured default profile for active SIM 0 is profile 1.


Data Connection Information
===========================
Profile 1, Packet Session Status = ACTIVE
        Cellular0/2/0:
        Data Packets Transmitted = 198 ,  Received = 209
        Data Transmitted = 14410 bytes, Received = 24882 bytes
        IP address = 29.29.29.196
        IPV6 address = 2001:2678:2680:5FD7:DDE7:70E1:DC07:CCB7/64  Scope: Global
        Primary DNS address = 8.0.0.8
        Secondary DNS address = 8.8.4.4
        Primary DNS IPV6 address = 2001:4860:4860:0:0:0:0:8888
        Secondary DNS IPV6 address = 2001:4860:4860:0:0:0:0:8844
Profile 2, Packet Session Status = ACTIVE
        Cellular0/2/1:
        Data Packets Transmitted = 12 ,  Received = 13
        Data Transmitted = 1200 bytes, Received = 1144 bytes
        IP address = 21.21.21.206
        IPV6 address = 2001:567A:567A:1480:5DD6:18D1:BD63:49DA/64  Scope: Global
        Primary DNS address = 171.70.168.183
        Secondary DNS address = 8.8.8.8
        Primary DNS IPV6 address = 2001:4860:4860:0:0:0:0:8888
        Secondary DNS IPV6 address = 2001:4860:4860:0:0:0:0:8844
Profile 3, Packet Session Status = INACTIVE
Profile 4, Packet Session Status = INACTIVE
Profile 5, Packet Session Status = INACTIVE
Profile 6, Packet Session Status = INACTIVE
Profile 7, Packet Session Status = INACTIVE
Profile 8, Packet Session Status = INACTIVE
Profile 9, Packet Session Status = INACTIVE
Profile 10, Packet Session Status = INACTIVE
Profile 11, Packet Session Status = INACTIVE
Profile 12, Packet Session Status = INACTIVE
Profile 13, Packet Session Status = INACTIVE
Profile 14, Packet Session Status = INACTIVE
Profile 15, Packet Session Status = INACTIVE
Profile 16, Packet Session Status = INACTIVE

Network Information
===================
Current System Time = Tue Jan 8 23:24:22 1980
```

```
 --More--
*Jun 19 06:13:14.665: %IOSXE_OIR-6-INSSPA: SPA inserted in sCurrent Service Status = Normal
Current Service = Packet switched
Current Roaming Status = Roaming
Network Selection Mode = Automatic
Network = 123 456
Mobile Country Code (MCC) = 123
Mobile Network Code (MNC) = 456
Packet switch domain(PS) state = Attached
LTE Carrier Aggregation state = Deconfigured
Registration state(EMM) = Registered
EMM Sub State = Normal Service
Tracking Area Code (TAC) = 1801
Cell ID = 768001
Network MTU is not Available

Radio Information
=================
Radio power mode = online
LTE Rx Channel Number =  2000
LTE Tx Channel Number =  20000
LTE Band =  4
LTE Bandwidth = 10 MHz
Current RSSI = -71 dBm
Current RSRP = -95 dBm
Current RSRQ = -7 dB
Current SNR = 26.4  dB
Physical Cell Id = 12
Number of nearby cells = 1
Idx      PCI (Physical Cell Id)
--------------------------------
1              12
Radio Access Technology(RAT) Preference = LTE
Radio Access Technology(RAT) Selected = LTE

Modem Security Information
==========================
Active SIM = 0
SIM switchover attempts = 0
Card Holder Verification (CHV1) = Disabled
SIM Status = OK
SIM User Operation Required = None
Number of CHV1 Retries remaining = 3

Cellular Firmware List
==========================
 Idx Carrier      FwVersion    PriVersion   Status
 1   ATT          02.20.03.00  002.019_000  Inactive
 2   GENERIC      02.20.03.00  002.017_000  Active
 3   SPRINT       02.20.03.22  002.020_000  Inactive
 4   TELSTRA      02.20.03.00  002.018_000  Inactive
 5   VERIZON      02.20.03.22  002.026_000  Inactive

Firmware Activation mode : AUTO

GPS Information
==========================

GPS Info
-------------
GPS Feature: enabled
GPS Mode Configured: not configured
GPS Status: NMEA Disabled
```

```
SMS Information
===============
Incoming Message Information
----------------------------
SMS stored in modem = 0
SMS archived since booting up = 0
Total SMS deleted since booting up = 0
Storage records allocated = 25
Storage records used = 0
Number of callbacks triggered by SMS = 0
Number of successful archive since booting up = 0
Number of failed archive since booting up = 0

Outgoing Message Information
----------------------------
Total SMS sent successfully = 0
Total SMS send failure = 0
Number of outgoing SMS pending = 0
Number of successful archive since booting up = 0
Number of failed archive since booting up = 0
Last Outgoing SMS Status = SUCCESS
Copy-to-SIM Status =     0x0
Send-to-Network Status = 0x0
Report-Outgoing-Message-Number:
  Reference Number =     0
  Result Code =          0x0
  Diag Code =            0x0 0x0 0x0 0x0 0x0

SMS Archive URL =

Error Information
=================

This command is not supported on 4G modems.


Modem Crashdump Information
===========================
Modem crashdump logging: off
```

Successful Call Setup

The following is a sample output when a call is set up. It shows a received IP address from the network. Call setup is successful and data path is open.

```
debug dialer
debug cellular 0/2/0 messages callcontrol
```

Modem Troubleshooting Using Integrated Modem DM Logging

As part of the 3G and 4G serviceability enhancement in Cisco IOS Release 15.2(4)M2 and Cisco IOS Release 15.3(1)T, DM log collection has been integrated into Cisco IOS, eliminating the need for an external PC and simplifying the DM log collection process. The lte modem dm-log command can be used in controller cellular configuration mode to configure integrated DM logging to monitor traffic on the modem. See the Cisco 3G and 4G Serviceability Enhancement User Guide for more information on configuring Integrated DM Logging parameters.

Modem Settings for North America and Carriers Operating on 700 MHz Band

For LTE-EA deployments in North America and for carriers operating in the 700 MHz band, the following changes to the modem settings are required to prevent long network attach times.

The output of show cellular x/x/x all command shows the following:

- Current RSSI is –125 dBM
- LTE Technology Preference = No preference specified (AUTO)

The following sections explain useful commands for changing modem settings:

Changing Modem Settings

To change the modem settings to force the modem to scan different technologies, use the following Cisco IOS command:

```
Router# cellular 0/2/0 lte technology ?
auto  Automatic LTE Technology Selection
  lte   LTE
  umts  UMTS
```

Electronic Serial Number (ESN)

The ESN number is located directly on the modem label in hexadecimal notation. It can also be retrieved using the Cisco IOS CLI using the show cellular *slot/port/module* **hardware** command.

The sample output below shows the ESN number:

```
Hardware Information
====================
Electronic Serial Number (ESN) = 0x603c9854 [09603971156]
Electronic Serial Number (ESN) = <specific ESN in hexadecimal> [specific ESN in decimal]
```

Additional References

Related Documents

Related Topic	Document Title
Hardware Overview and Installation	• *Cisco 4G-LTE Wireless WAN EHWIC* http://www.cisco.com/en/US/docs/routers/access/interfaces/ic/hardware/installation/guide/EHWIC-4G-L • *Cisco Fourth-Generation LTE Network Interface Module Installation Guide* http://www.cisco.com/c/en/us/td/docs/routers/access/interfaces/NIM/hardware/installation/guide/4GLTEN

Related Topic	Document Title
Supported Cisco antennas and cables	• *Installing Cisco Interface Cards in Cisco Access Routers* http://www.cisco.com/en/US/docs/routers/access/interfaces/ic/hardware/installation/guide/inst_ic.html • *Cisco 4G/3G Omnidirectional Dipole Antenna (4G-LTE-ANTM-D)* http://www.cisco.com/en/US/docs/routers/access/wireless/hardware/notes/4G3G_ant.html • *Cisco 4G Indoor Ceiling-Mount Omnidirectional Antenna (4G-ANTM-OM-CM)* http://www.cisco.com/en/US/docs/routers/access/wireless/hardware/notes/antcm4gin.html • *Cisco Outdoor Omnidirectional Antenna for 2G/3G/4G Cellular (ANT-4G-OMNI-OUT-N)* http://www.cisco.com/en/US/docs/routers/connectedgrid/antennas/installing/Outdoor_Omni_for_2G_3G_4G • *Cisco Integrated 4G Low-Profile Outdoor Saucer Antenna (ANT-4G-SR-OUT-TNC)* http://www.cisco.com/en/US/docs/routers/connectedgrid/antennas/installing/4G_LowProfile_Outdoor_Sauce • *Cisco Single-Port Antenna Stand for Multiband TNC Male-Terminated Portable Antenna (Cisco 4G-AE* http://www.cisco.com/en/US/docs/routers/access/wireless/hardware/notes/4Gantex15-10r.html • *Cisco 4G Lightning Arrestor (4G-ACC-OUT-LA)* http://www.cisco.com/en/US/docs/routers/access/wireless/hardware/notes/4Glar.html • *Lightning Arrestor for the Cisco 1240 Connected Grid Router* http://www.cisco.com/en/US/docs/routers/connectedgrid/lightning_arrestor/Lightning_Arrestor_for_the_Cisc Cisco 4G Indoor/Outdoor Active GPS Antenna (GPS-ACT-ANTM-SMA)
Datasheet	• Modules data sheets for ISR4k http://www.cisco.com/c/en/us/products/routers/4000-series-integrated-services-routers-isr/datasheet-listing.h • LTE datasheet http://www.cisco.com/en/US/docs/routers/access/wireless/hardware/notes/4Gantex15-10r.html http://www.cisco.com/c/en/us/td/docs/routers/access/4400/roadmap/isr4400roadmap.html

MIBs

MIB	MIBs Link
• IF-MIB • CISCO-ENTITY-VENDORTYPE-OID-MIB • CISCO-WAN-3G-MIB	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 3025	Mobile IP Vendor/Organization-Specific Extensions

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

CHAPTER 32

Configuration Examples

This chapter provides examples of configuring common networking tasks on the router. The examples in this chapter are provided for illustrative purposes only; little or no context is given with these examples. For more information, see Installing the Software, on page 93.

When reading this section, also be aware that networking configurations are complex and can be configured in many ways. The examples in this section show one method of accomplishing a configuration.

This chapter contains the following examples:

Copying the Consolidated Package from the TFTP Server to the Router

The following example shows how to copy the consolidated package from the TFTP server to the router:

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul  2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 17:48:41 +00:00  .prst_sync
324481  drwx        4096   Jul  2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul  2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 17:49:14 +00:00  tracelogs
32449  drwx        4096   Jul  2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul  2 2012 15:27:08 +00:00  vman_fdb

7451738112 bytes total (7015186432 bytes free)
Router# copy tftp bootflash:
Address or name of remote host []? 10.81.116.4
Source filename []? rtp-isr4400-54/isr4400.bin
```

```
Destination filename [isr4400.bin]?
Accessing tftp://10.81.116.4/rtp-isr4400-54/isr4400.bin...
Loading rtp-isr4400-54/isr4400.bin from 10.81.116.4 (via GigabitEthernet0): !!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 424317088 bytes]

424317088 bytes copied in 371.118 secs (1143348 bytes/sec)
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 17:48:41 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:05:07 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
   13  -rw-   424317088  Sep 13 2012 18:01:41 +00:00  isr4400.bin

7451738112 bytes total (6590910464 bytes free)
```

Configuring the Router to Boot Using the Consolidated Package Stored on the Router

The following example shows how to configure the router to boot using the consolidated package stored on the router:

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 17:48:41 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:05:07 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
   13  -rw-   424317088  Sep 13 2012 18:01:41 +00:00  isr4400.bin

7451738112 bytes total (6590910464 bytes free)


Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# boot system bootflash:isr4400.bin
Router(config)# config-register 0x2102
Router(config)# exit
Router# show run | include boot
boot-start-marker
boot system bootflash:isr4400.bin
boot-end-marker
license boot level adventerprise
Router# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

```
Router# reload
Proceed with reload? [confirm]
Sep 13 18:08:36.311 R0/0: %PMAN-5-EXITACTION: Process manager is exiting: process exit
with reload chassis code


Initializing Hardware ...

System integrity status: c0000600
Failures detected:
 Boot FPGA corrupt

Key Sectors:(Primary,GOOD),(Backup,GOOD),(Revocation,GOOD)
Size of Primary = 2288 Backup = 2288 Revocation = 300

ROM:RSA Self Test Passed
ROM:Sha512 Self Test Passed
Self Tests Latency: 58 msec


System Bootstrap, Version 12.2(20120618:163328) [username-ESGROM_20120618_GAMMA 101],
DEVELOPMENT SOFTWARE
Copyright (c) 1994-2012  by cisco Systems, Inc.
Compiled Mon 06/18/2012 12:39:32.05 by username

Current image running: Boot ROM0

Last reset cause: LocalSoft

Cisco ISR 4400 platform with 4194304 Kbytes of main memory

File size is 0x194a90a0
Located isr4400.bin
Image size 424317088 inode num 13, bks cnt 103594 blk size 8*512
###########################################################################################
Boot image size = 424317088 (0x194a90a0) bytes

ROM:RSA Self Test Passed
ROM:Sha512 Self Test Passed
Self Tests Latency: 58 msec

Package header rev 1 structure detected
Calculating SHA-1 hash...done
validate_package: SHA-1 hash:
 calculated 7294dffc:892a6c35:a7a133df:18c032fc:0670b303
 expected   7294dffc:892a6c35:a7a133df:18c032fc:0670b303
Signed Header Version Based Image Detected

Using FLASH based Keys of type = PRIMARY KEY STORAGE
Using FLASH based Keys of type = ROLLOVER KEY STORAGE
RSA Signed DEVELOPMENT Image Signature Verification Successful.
Package Load Test Latency : 5133 msec
Image validated
%IOSXEBOOT-4-BOOT_ACTIVITY_LONG_TIME: (local/local): load_modules took: 2 seconds, expected

max time 2 seconds

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
```

```
Software clause at DFARS sec. 252.227-7013.

           cisco Systems, Inc.
           170 West Tasman Drive
           San Jose, California 95134-1706


Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version

15.3(20120910:013018) [mcp_dev-BLD-BLD_MCP_DEV_LATEST_20120910_000023-ios 153]
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Sun 09-Sep-12 21:28 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2012 by cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

Warning:  the compile-time code checksum does not appear to be present.
cisco ISR4451/K9 (2RU) processor with 1133589K/6147K bytes of memory.
Processor board ID FGL1619100P
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of Compact flash at bootflash:.
7816688K bytes of USB flash at usb0:.

Press RETURN to get started!
```

Extracting the Subpackages from a Consolidated Package into the Same File System

The following example shows how to extract the subpackages from a consolidated package into the same file system.

After entering the **request platform software package expand file bootflash:isr4400.bin** command (note that the **to** option is not used) the subpackages are extracted from the consolidated package into **bootflash:**

```
Router> enable
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:12:58 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:13:31 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
   13  -rw-   424317088  Sep 13 2012 18:01:41 +00:00  isr4400.bin

7451738112 bytes total (6590029824 bytes free)
Router# request platform software package expand file bootflash:isr4400.bin
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:12:58 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:16:49 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
   13  -rw-   424317088  Sep 13 2012 18:01:41 +00:00  isr4400.bin
778756  -rw-   112911096  Sep 13 2012 18:15:49 +00:00
isr4400-espbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778757  -rw-     2220784  Sep 13 2012 18:15:49 +00:00
isr4400-firmware_dsp_sp2700.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778758  -rw-      371440  Sep 13 2012 18:15:49 +00:00
isr4400-firmware_fpge.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778759  -rw-     8080112  Sep 13 2012 18:15:49 +00:00
isr4400-firmware_nim_t1e1.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778760  -rw-     9331440  Sep 13 2012 18:15:49 +00:00
isr4400-firmware_sm_1t3e3.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778761  -rw-      379632  Sep 13 2012 18:15:49 +00:00
isr4400-firmware_ucse.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
 --More--          778754  -rw-       10540  Sep 13 2012 18:15:48 +00:00
isr4400-packages-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.conf
778762  -rw-    27218680  Sep 13 2012 18:15:50 +00:00
isr4400-rpaccess.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778763  -rw-    78938264  Sep 13 2012 18:15:50 +00:00
isr4400-rpbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778764  -rw-    45177592  Sep 13 2012 18:15:50 +00:00
isr4400-rpcontrol.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778765  -rw-   114662144  Sep 13 2012 18:16:01 +00:00
isr4400-rpios-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778766  -rw-    26360568  Sep 13 2012 18:16:03 +00:00
isr4400-sipbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778767  -rw-    13091576  Sep 13 2012 18:16:06 +00:00
isr4400-sipspa.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
778755  -rw-       11349  Sep 13 2012 18:16:06 +00:00  packages.conf

7451738112 bytes total (6150725632 bytes free)
```

Extracting the Subpackages from a Consolidated Package into a Different File System

The following example shows how to extract the subpackages from a consolidated package into a different file system.

The initial **dir usb0:** command shows that there are no subpackages in the **bootflash:** directory.

After the **request platform software package expand file usb0:isr4400.bin to bootflash:** command is entered, the subpackages are displayed in the **bootflash:** directory. The isr4400.bin consolidated package file is in the **usb0:** directory.

```
Router# dir usb0:
Directory of usb0:/

  121  -rwx   424317088  Sep 13 2012 18:27:50 +00:00  isr4400.bin

7988666368 bytes total (7564341248 bytes free)

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:12:58 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:41:51 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb

7451738112 bytes total (6590418944 bytes free)
Router# request platform software package expand file usb0:isr4400.bin to bootflash:
Verifying parameters
Validating package type
Copying package files
SUCCESS: Finished expanding all-in-one software package.
Router# dir bootflash:
Directory of bootflash:/
11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:12:58 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:46:52 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
454276  -rw-   112911096  Sep 13 2012 18:46:05 +00:00
isr4400-espbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454277  -rw-     2220784  Sep 13 2012 18:46:05 +00:00
isr4400-firmware_dsp_sp2700.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454278  -rw-      371440  Sep 13 2012 18:46:05 +00:00
isr4400-firmware_fpge.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454279  -rw-     8080112  Sep 13 2012 18:46:05 +00:00
isr4400-firmware_nim_t1e1.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454280  -rw-     9331440  Sep 13 2012 18:46:06 +00:00
isr4400-firmware_sm_1t3e3.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454281  -rw-      379632  Sep 13 2012 18:46:06 +00:00
```

```
isr4400-firmware_ucse.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
 --More--         454274  -rw-       10540  Sep 13 2012 18:46:05 +00:00
isr4400-packages-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.conf
454282  -rw-    27218680  Sep 13 2012 18:46:06 +00:00
isr4400-rpaccess.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454283  -rw-    78938264  Sep 13 2012 18:46:06 +00:00
isr4400-rpbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454284  -rw-    45177592  Sep 13 2012 18:46:06 +00:00
isr4400-rpcontrol.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454285  -rw-   114662144  Sep 13 2012 18:46:16 +00:00
isr4400-rpios-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454286  -rw-    26360568  Sep 13 2012 18:46:19 +00:00
isr4400-sipbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454287  -rw-    13091576  Sep 13 2012 18:46:21 +00:00
isr4400-sipspa.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454275  -rw-       11349  Sep 13 2012 18:46:21 +00:00  packages.conf

7451738112 bytes total (6575869952 bytes free)
```

Configuring the Router to Boot Using Subpackages

After placing the provisioning file and subpackage files in a directory and booting the router, we recommend that you do not rename, delete, or alter any of these files. Renaming, deleting, or altering the files can lead to unpredictable router problems and behaviors. Each version of a consolidated package contains subpackages that are similar to those shown in the following table. However, each version of a consolidated package may contain different versions of each subpackage.

Table 52: Subpackages

Subpackage	Description
RPBase	Provides the operating system software for the Route Processor. This is the only bootable package.
RPControl	Controls the control plane processes that act as the interface between the Cisco IOS process and the rest of the platform.
RPAccess	Exports processing of restricted components, such as Secure Socket Layer (SSL), Secure Shell (SSH), and other security features.
RPIOS	Provides the Cisco IOS kernel, where Cisco IOS XE features are stored and run. Each consolidated package has a different version of RPIOS.
ESPBase	Provides the Embedded Services Processor (ESP) operating system and control processes, and ESP software.
SIPBase	Provides control processes.
SIPSPA	Provides Input/Output (I/O) drivers.
Firmware	Firmware subpackage. The name of the subpackage includes the module type, which either refers to a Network Information Module (NIM) or Cisco Enhanced Service Module.

The following example shows how to configure the router to boot using subpackages:

The **dir bootflash:** command confirms that all subpackages and the provisioning file are in the same file system, as shown in the following example:

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:12:58 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:46:52 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
454276  -rw-   112911096  Sep 13 2012 18:46:05 +00:00
isr4400-espbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454277  -rw-     2220784  Sep 13 2012 18:46:05 +00:00
isr4400-firmware_dsp_sp2700.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454278  -rw-      371440  Sep 13 2012 18:46:05 +00:00
isr4400-firmware_fpge.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454279  -rw-     8080112  Sep 13 2012 18:46:05 +00:00
isr4400-firmware_nim_t1e1.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454280  -rw-     9331440  Sep 13 2012 18:46:06 +00:00
isr4400-firmware_sm_1t3e3.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454281  -rw-      379632  Sep 13 2012 18:46:06 +00:00
isr4400-firmware_ucse.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
 --More--         454274  -rw-       10540  Sep 13 2012 18:46:05 +00:00
isr4400-packages-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.conf
454282  -rw-    27218680  Sep 13 2012 18:46:06 +00:00
isr4400-rpaccess.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454283  -rw-    78938264  Sep 13 2012 18:46:06 +00:00
isr4400-rpbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454284  -rw-    45177592  Sep 13 2012 18:46:06 +00:00
isr4400-rpcontrol.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454285  -rw-   114662144  Sep 13 2012 18:46:16 +00:00
isr4400-rpios-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454286  -rw-    26360568  Sep 13 2012 18:46:19 +00:00
isr4400-sipbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454287  -rw-    13091576  Sep 13 2012 18:46:21 +00:00
isr4400-sipspa.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454275  -rw-       11349  Sep 13 2012 18:46:21 +00:00  packages.conf

7451738112 bytes total (6575869952 bytes free)

Router# show running | include boot
boot-start-marker
boot-end-marker
license boot level adventerprise
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# boot system bootflash:packages.conf
Router(config)# config-register 0x2102
Router(config)# exit
Router# show running | include boot
boot-start-marker
boot system bootflash:packages.conf
boot-end-marker
license boot level adventerprise
Router# copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
Router# reload
```

```
Proceed with reload? [confirm]
Sep 13 18:49:39.720 R0/0: %PMAN-5-EXITACTION: Process manager is exiting: process exit with
 reload chassis code


Initializing Hardware ...

System integrity status: c0000600
Failures detected:
 Boot FPGA corrupt


Key Sectors:(Primary,GOOD),(Backup,GOOD),(Revocation,GOOD)
Size of Primary = 2288 Backup = 2288 Revocation = 300

ROM:RSA Self Test Passed
ROM:Sha512 Self Test Passed
Self Tests Latency: 58 msec


System Bootstrap, Version 12.2(20120618:163328) [username-ESGROM_20120618_GAMMA 101],
DEVELOPMENT SOFTWARE
Copyright (c) 1994-2012  by cisco Systems, Inc.
Compiled Mon 06/18/2012 12:39:32.05 by username

Current image running: Boot ROM0

Last reset cause: LocalSoft

Cisco ISR 4400 platform with 4194304 Kbytes of main memory

File size is 0x00002c55
Located packages.conf
Image size 11349 inode num 454275, bks cnt 3 blk size 8*512
#
File size is 0x04b48098
Located isr4400-rpbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
Image size 78938264 inode num 454283, bks cnt 19273 blk size 8*512
########################################################################################################################################################################################################################################################################################################################################################################
Boot image size = 78938264 (0x4b48098) bytes

ROM:RSA Self Test Passed
ROM:Sha512 Self Test Passed
Self Tests Latency: 58 msec

Package header rev 1 structure detected
Calculating SHA-1 hash...done
validate_package: SHA-1 hash:
 calculated dbe960a6:d239245c:76d93622:d6c31a41:40e9e420
 expected   dbe960a6:d239245c:76d93622:d6c31a41:40e9e420
Signed Header Version Based Image Detected

Using FLASH based Keys of type = PRIMARY KEY STORAGE
Using FLASH based Keys of type = ROLLOVER KEY STORAGE
RSA Signed DEVELOPMENT Image Signature Verification Successful.
Package Load Test Latency : 1159 msec
Image validated

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
```

```
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

           cisco Systems, Inc.
           170 West Tasman Drive
           San Jose, California 95134-1706



Cisco IOS Software, IOS-XE Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version
 15.3(20120910:013018) [mcp_dev-BLD-BLD_MCP_DEV_LATEST_20120910_000023-ios 153]
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Sun 09-Sep-12 21:28 by mcpre



Cisco IOS-XE software, Copyright (c) 2005-2012 by cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.



This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

Warning:  the compile-time code checksum does not appear to be present.
cisco ISR4451/K9 (2RU) processor with 1133589K/6147K bytes of memory.
Processor board ID FGL1619100P
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of Compact flash at bootflash:.
7816688K bytes of USB flash at usb0:.

Press RETURN to get started!

Router>
Router> en
Router# show version
Cisco IOS XE Software, Version BLD_V154_3_S_XE313_THROTTLE_LATEST_20140527_070027-ext
Cisco IOS Software, ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Experimental Version
15.4(20140527:095327)
[v154_3_s_xe313_throttle-BLD-BLD_V154_3_S_XE313_THROTTLE_LATEST_20140527_070027-ios 156]


IOS XE Version: BLD_V154_3_S_XE313_THROTTLE_LATEST
```

```
Cisco IOS-XE software, Copyright (c) 2005-2014 by cisco Systems, Inc.
All rights reserved.  Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0.  The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY.  You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0.  For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.


ROM: IOS-XE ROMMON

Router uptime is 1 minute
Uptime for this control processor is 4 minutes
 --More--         System returned to ROM by reload
System image file is "bootflash:packages.conf"
Last reload reason: Reload Command

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

License Level: adventerprise
License Type: EvalRightToUse
 --More--         Next reload license Level: adventerprise

cisco ISR4451/K9 (2RU) processor with 1133589K/6147K bytes of memory.
Processor board ID FGL1619100P
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7393215K bytes of Compact flash at bootflash:.
7816688K bytes of USB flash at usb0:.

Configuration register is 0x2102

Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:53:29 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:54:03 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
454276  -rw-   112911096  Sep 13 2012 18:46:05 +00:00
isr4400-espbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454277  -rw-     2220784  Sep 13 2012 18:46:05 +00:00
```

```
isr4400-firmware_dsp_sp2700.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454278  -rw-      371440  Sep 13 2012 18:46:05 +00:00
isr4400-firmware_fpge.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454279  -rw-     8080112  Sep 13 2012 18:46:05 +00:00
isr4400-firmware_nim_t1e1.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454280  -rw-     9331440  Sep 13 2012 18:46:06 +00:00
isr4400-firmware_sm_1t3e3.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454281  -rw-      379632  Sep 13 2012 18:46:06 +00:00
isr4400-firmware_ucse.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
 --More--         454274  -rw-       10540  Sep 13 2012 18:46:05 +00:00
isr4400-packages-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.conf
454282  -rw-    27218680  Sep 13 2012 18:46:06 +00:00
isr4400-rpaccess.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454283  -rw-    78938264  Sep 13 2012 18:46:06 +00:00
isr4400-rpbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454284  -rw-    45177592  Sep 13 2012 18:46:06 +00:00
isr4400-rpcontrol.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454285  -rw-   114662144  Sep 13 2012 18:46:16 +00:00
isr4400-rpios-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454286  -rw-    26360568  Sep 13 2012 18:46:19 +00:00
isr4400-sipbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454287  -rw-    13091576  Sep 13 2012 18:46:21 +00:00
isr4400-sipspa.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg
454275  -rw-       11349  Sep 13 2012 18:46:21 +00:00  packages.conf

7451738112 bytes total (6574940160 bytes free)

Router# del isr4400*
Delete filename [isr4400*]?
Delete bootflash:/isr4400-espbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg? [confirm]
Delete bootflash:/isr4400-firmware_dsp_sp2700.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg?
[confirm]
Delete bootflash:/isr4400-firmware_fpge.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg? [confirm]
Delete bootflash:/isr4400-firmware_nim_t1e1.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg?
[confirm]
Delete bootflash:/isr4400-firmware_sm_1t3e3.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg?
[confirm]
Delete bootflash:/isr4400-firmware_ucse.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg? [confirm]
Delete bootflash:/isr4400-packages-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.conf?
[confirm]
Delete bootflash:/isr4400-rpaccess.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg? [confirm]
Delete bootflash:/isr4400-rpbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg? [confirm]
Delete bootflash:/isr4400-rpcontrol.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg? [confirm]
Delete bootflash:/isr4400-rpios-universalk9.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg?
[confirm]
Delete bootflash:/isr4400-sipbase.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg? [confirm]
Delete bootflash:/isr4400-sipspa.BLD_MCP_DEV_LATEST_20120910_000023.SSA.pkg? [confirm]
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:53:29 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 18:54:03 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
454275  -rw-       11349  Sep 13 2012 18:46:21 +00:00  packages.conf

7451738112 bytes total (6574952448 bytes free)
Router# del packages.conf
Delete filename [packages.conf]?
```

```
Delete bootflash:/packages.conf? [confirm]
Router# copy tftp bootflash:
Address or name of remote host []? 10.81.116.4
Source filename []? rtp-isr4400-54/isr4400.bin
Destination filename [isr4400.bin]?
Accessing tftp://10.81.116.4/rtp-isr4400-54/isr4400.bin...
Loading rtp-isr4400-54/isr4400.bin from 10.81.116.4 (via GigabitEthernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 424317088 bytes]

424317088 bytes copied in 351.758 secs (1206276 bytes/sec)
```

Backing Up Configuration Files

This section provides the following examples:

- Copying a Startup Configuration File to BootFlash, on page 567
- Copying a Startup Configuration File to a USB Flash Drive, on page 568
- Copying a Startup Configuration File to a TFTP Server, on page 568

Copying a Startup Configuration File to BootFlash

```
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:53:29 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 19:03:19 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
   13  -rw-   424317088  Sep 13 2012 19:02:50 +00:00  isr4400.bin

7451738112 bytes total (6150721536 bytes free)
Router# copy nvram:startup-config bootflash:
Destination filename [startup-config]?
1367 bytes copied in 0.116 secs (11784 bytes/sec)
Router# dir bootflash:
Directory of bootflash:/

   11  drwx       16384   Jul 2 2012 15:25:23 +00:00  lost+found
16225  drwx        4096  Jul 31 2012 19:30:48 +00:00  core
178465  drwx        4096  Sep 13 2012 18:53:29 +00:00  .prst_sync
324481  drwx        4096   Jul 2 2012 15:26:54 +00:00  .rollback_timer
   12  -rw-           0   Jul 2 2012 15:27:06 +00:00  tracelogs.696
373153  drwx      114688  Sep 13 2012 19:03:19 +00:00  tracelogs
32449  drwx        4096   Jul 2 2012 15:27:08 +00:00  .installer
681409  drwx        4096  Jul 31 2012 19:15:39 +00:00  .ssh
697633  drwx        4096   Jul 2 2012 15:27:08 +00:00  vman_fdb
   13  -rw-   424317088  Sep 13 2012 19:02:50 +00:00  isr4400.bin
   14  -rw-        1367  Sep 13 2012 19:03:57 +00:00  startup-config
```

```
7451738112 bytes total (6150717440 bytes free)
Router# copy bootflash:startup-config tftp:
Address or name of remote host []? 172.18.40.33
Destination filename [router-confg]? startup-config
!!
1367 bytes copied in 0.040 secs (34175 bytes/sec)
Router# exit

Router con0 is now available


Press RETURN to get started.
```

Copying a Startup Configuration File to a USB Flash Drive

```
Router# dir usb0:
Directory of usb0:/

No files in directory

4094840832 bytes total (4094836736 bytes free)
Router# copy nvram:startup-config usb0:
Destination filename [startup-config]?
1644 bytes copied in 0.248 secs (6629 bytes/sec)
Router# dir usb0:
Directory of usb0:/

3097__-rwx________1644__ Oct 3 2012 14:53:50 +00:00__startup-config

4094840832 bytes total (4094832640 bytes free)
Router#
```

Copying a Startup Configuration File to a TFTP Server

```
Router# copy nvram:startup-config tftp:
Address or name of remote host []? 172.18.40.4
Destination filename [router-confg]?
!!
3274 bytes copied in 0.039 secs (83949 bytes/sec)
Router#
```

Displaying Digitally Signed Cisco Software Signature Information

In this example, authenticity details for a consolidated package are displayed on the screen:

```
router# show software authenticity running
PACKAGE isr4400-rpbase.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
-------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48E17
```

```
    Hash Algorithm            : SHA512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A

    Verifier Information
        Verifier Name         : rp_base
        Verifier Version      : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-rpcontrol.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
----------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48DA3
    Hash Algorithm            : SHA512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A

    Verifier Information
        Verifier Name         : rp_base
        Verifier Version      : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-rpios-universalk9.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
------------------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48E98
    Hash Algorithm            : SHA512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A

    Verifier Information
        Verifier Name         : rp_base
        Verifier Version      : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-rpaccess.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
---------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48DB4
    Hash Algorithm            : SHA512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A

    Verifier Information
        Verifier Name         : rp_base
        Verifier Version      : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-firmware_dsp_sp2700.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
--------------------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48DBE
```

```
    Hash Algorithm            : SHA512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A

    Verifier Information
        Verifier Name         : rp_base
        Verifier Version      : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-firmware_sm_1t3e3.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
-------------------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48DC7
    Hash Algorithm            : SHA512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A

    Verifier Information
        Verifier Name         : rp_base
        Verifier Version      : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-firmware_nim_t1e1.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
---------------------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48D74
    Hash Algorithm            : SHA512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A

    Verifier Information
        Verifier Name         : rp_base
        Verifier Version      : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-espbase.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
---------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48D64
    Hash Algorithm            : SHA512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A

    Verifier Information
        Verifier Name         : rp_base
        Verifier Version      : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-sipbase.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
---------------------------------------------------------------------
Image type                    : Special
    Signer Information
        Common Name           : CiscoSystems
        Organization Unit     : IOS-XE
        Organization Name     : CiscoSystems
    Certificate Serial Number : 50F48D94
```

```
    Hash Algorithm             : SHA512
    Signature Algorithm        : 2048-bit RSA
    Key Version                : A

    Verifier Information
        Verifier Name          : rp_base
        Verifier Version       : BLD_MCP_DEV_LATEST_20130114_162711

PACKAGE isr4400-sipspa.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
-------------------------------------------------------------------
Image type                     : Special
    Signer Information
        Common Name            : CiscoSystems
        Organization Unit      : IOS-XE
        Organization Name      : CiscoSystems
    Certificate Serial Number : 50F48D7F
    Hash Algorithm             : SHA512
    Signature Algorithm        : 2048-bit RSA
    Key Version                : A

    Verifier Information
        Verifier Name          : rp_base
        Verifier Version       : BLD_MCP_DEV_LATEST_20130114_162711

SYSTEM IMAGE
------------
Image type                     : Special
    Signer Information
        Common Name            : CiscoSystems
        Organization Unit      : IOS-XE
        Organization Name      : CiscoSystems
    Certificate Serial Number : 50F48F33
    Hash Algorithm             : SHA512
    Signature Algorithm        : 2048-bit RSA
    Key Version                : A

    Verifier Information
        Verifier Name          : ROMMON
        Verifier Version       : System Bootstrap, Version 12.2(20121015:145923
ROMMON
------
Image type                     : Special
    Signer Information
        Common Name            : CiscoSystems
        Organization Unit      : IOS-XE
        Organization Name      : CiscoSystems
    Certificate Serial Number : 50801108
    Hash Algorithm             : SHA512
    Signature Algorithm        : 2048-bit RSA
    Key Version                : A

    Verifier Information
        Verifier Name          : ROMMON
        Verifier Version       : System Bootstrap, Version 12.2(20121015:145923
Microloader
-----------
Image type                     : Release
    Signer Information
        Common Name            : CiscoSystems
        Organization Name      : CiscoSystems
    Certificate Serial Number : bace997bdd9882f8569e5b599328a448
    Hash Algorithm             : HMAC-SHA256
    Verifier Information
        Verifier Name          : Hardware Anchor
```

```
Verifier Version      : F01001R06.02c4c06f82012-09-17
```

Obtaining the Description of a Module or Consolidated Package

In this example, internal details of the consolidated package are displayed on the screen:

```
router# request platform software package describe file
bootflash:isr4400-rpbase.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
Package: isr4400-rpbase.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg
  Size: 79755832
  Timestamp: 2013-01-15 15:46:59 UTC
  Canonical path: /bootflash/isr4400-rpbase.BLD_MCP_DEV_LATEST_20130114_162711.SSA.pkg

  Raw disk-file SHA1sum:
    5cd5916a216b147e3d9e33c0dc5afb18d86bda94

  Digital Signature Verified
  Computed SHA1sum:
    de80d5920819d224113b81a1d64b17449859952e
  Contained SHA1sum:
    de80d5920819d224113b81a1d64b17449859952e
  Hashes match. Package is valid.

  Header size:     760 bytes
  Package type:    30001
  Package flags:   0
  Header version:  1

  Internal package information:
    Name: rp_base
    BuildTime: 2013-01-14_14.55
    ReleaseDate: Mon-14-Jan-13-16:27
    BootArchitecture: i686
    RouteProcessor: overlord
    Platform: ISR
    User: mcpre
    PackageName: rpbase
    Build: BLD_MCP_DEV_LATEST_20130114_162711
    CardTypes:

  Package is bootable on RP when specified
  by packages provisioning file.
```

CHAPTER 33

Troubleshooting

System Report

System reports or crashinfo files save information that helps Cisco technical support representatives to debug problems that caused the Cisco IOS image to crash. It is necessary to collect critical crash information quickly and reliably and bundle it in a way that it can be identified with a specific crash occurrence. System reports are generated and saved into the '/core' directory, either on harddisk: or flash: filesystem. The system does not generate reports in case of a reload.

In case of a system crash, the following details are collected:

1. Full process core
   - IOSd core file and IOS crashinfo file if there was an IOSd process crash
2. Tracelogs
3. System process information
4. Bootup logs
5. Certain types of /proc information

This report is generated before the router goes down to rommon/bootloader. The information is stored in separate files which are then archived and compressed into the tar.gz bundle. This makes it convenient to get a crash snapshot in one place, and can be then moved off the box for analysis.

Device hostname, the ID of the module that generated the system report and its creation timestamp are embedded in the file name:

<hostname>_<moduleID>-system-report_<timestamp>.tar.gz

Example:

Router1_RP_0-system-report_20210204-163559-UTC

A device with hostname Router1 experienced an unexpected reload of RP0 module and the system-report was generated on 4th February 2021 at 4:39:59 PM UTC.

```
├── bootflash/
│   └── pd_info/
│       ├── dmesg_output-20210204-163538-UTC.log
│       ├── filesystems-20210204-163538-UTC.log
│       ├── memaudit-20210204-163538-UTC.log
│       ├── proc_cpuinfo-20210204-163538-UTC.log
│       ├── proc_diskstats-20210204-163538-UTC.log
│       ├── proc_interrupts-20210204-163538-UTC.log
│       ├── proc_oom_stats-20210204-163538-UTC.log
│       ├── proc_softirqs-20210204-163538-UTC.log
│       ├── system_report_trigger.log
│       └── top_output-20210204-163538-UTC.log
├── harddisk/
│   ├── core/
│   │   └── Router1_RP_0_hman_17716_20210212-123836-UTC.core.gz
│   └── tracelogs/
├── tmp/
│   ├── fp/
│   │   └── trace/
│   ├── maroon_stats/
│   ├── rp/
│   │   └── trace/
│   └── Router1_RP_0-bootuplog-20210204-163559-UTC.log
└── var/
    └── log/
        └── audit/
            └── audit.log
```

APPENDIX A

Unsupported Commands

The Cisco 4000 Series routers contain a series of commands with the **logging** or **platform** keywords that either produce no output or produce output that is not useful for customer purposes. Such commands that are not useful for customer purposes are considered as unsupported commands. You will not find any further Cisco documentation for the unsupported commands.

The following is a list of unsupported commands for the Cisco 4000 Series routers:

- clear logging onboard slot f0 dram
- clear logging onboard slot f0 voltage
- clear logging onboard slot f0 temperature
- show logging onboard slot f0 dram
- show logging onboard slot f0 serdes
- show logging onboard slot f0 status
- show logging onboard slot f0 temperature
- show logging onboard slot f0 uptime
- show logging onboard slot f0 uptime latest
- show logging onboard slot f0 voltage
- show logging onboard slot 0 dram
- show logging onboard slot 0 serdes
- show logging onboard slot 0 status
- show logging onboard slot 0 temperature
- show logging onboard slot 0 uptime
- show logging onboard slot 0 uptime latest
- show logging onboard slot 0 voltage
- show platform software adjacency r0 special
- show platform software adjacency rp active special

- show platform software ethernet rp active l2cp
- show platform software ethernet rp active l2cp interface GigabitEthernet0
- show platform software ethernet rp active loopback
- show platform software ethernet rp active vfi
- show platform software ethernet r0 vfi
- show platform software ethernet r0 vfi id 0
- show platform software ethernet r0 vfi name GigabitEthernet0
- show platform software ethernet r0 l2cp
- show platform software ethernet r0 l2cp interface GigabitEthernet0
- show platform software ethernet r0 bridge-domain statistics
- show platform software flow r0 exporter name GigabitEthernet0
- show platform software flow r0 exporter statistics
- show platform software flow r0 global
- show platform software flow r0 flow-def
- show platform software flow r0 interface
- show platform software flow r0 ios
- show platform software flow r0 monitor
- show platform software flow r0 sampler
- show platform hardware qfp active classification feature-manager label GigabitEthernet 0 0
- show platform software interface f0 del-track
- show platform software interface fp active del-track
- show platform software rg r0 services
- show platform software rg r0 services rg-id 0
- show platform software rg r0 services rg-id 0 verbose
- show platform software rg r0 services verbose
- show platform software rg r0 statistics
- show platform software rg rp active services
- show platform software rg rp active services rg-id 0
- show platform software rg rp active services rg-id 0 verbose
- show platform software rg rp active statistics
- show platform hardware slot 0 dram statistics
- show platform hardware slot f0 dram statistics

- show platform hardware slot 0 eobc interface primary rmon
- show platform hardware slot 0 eobc interface primary status
- show platform hardware slot 0 eobc interface standby rmon
- show platform hardware slot 0 eobc interface standby status
- show platform hardware slot f0 eobc interface primary rmon
- show platform hardware slot f0 eobc interface primary status
- show platform hardware slot f0 eobc interface standby rmon
- show platform hardware slot f0 eobc interface standby status
- show platform hardware slot f0 sensor consumer
- show platform hardware slot f0 sensor producer