Cisco IOS XR System Management Configuration Guide for the Cisco CRS Router, Release 6.1.x

First Published: 2016-11-01

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

CONTENTS

Preface

This guide describes the System Management configuration details for Cisco IOS XR software. This chapter contains details on the changes made to this document.

Changes to This Document

This table lists the technical changes made to this document since it was first released.

Table 1: Changes to This Document

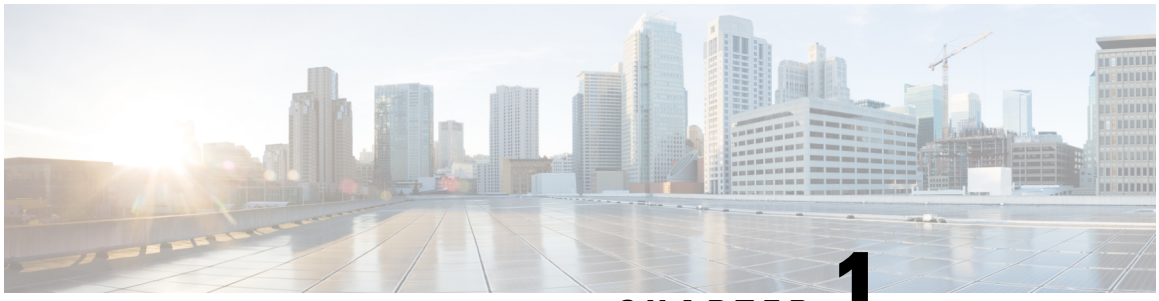
Date	Summary
November 2016	Initial release of this document.

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 Marketplace.
- 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.



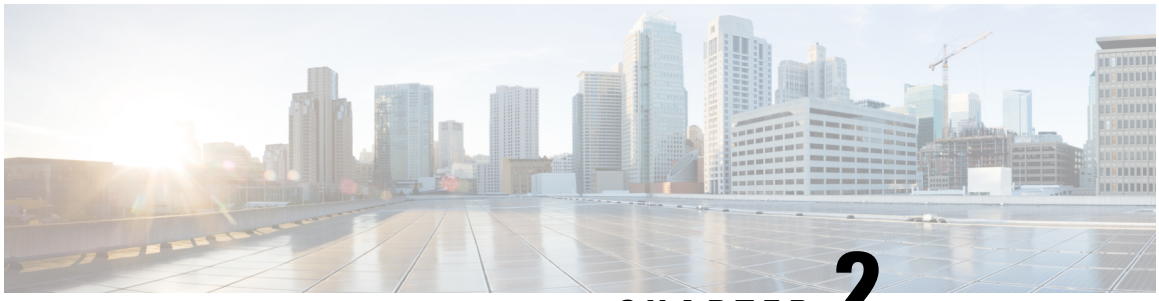
CHAPTER 1

New and Changed System Management Features

This chapter lists all the features that have been added or modified in this guide. The table also contains references to these feature documentation sections.

System Management Features Added or Modified in IOS XR Release 6.1.x

Feature	Description	Changed in Release	Where Documented
None	No new features introduced	Not applicable	Not applicable



CHAPTER 2

Configuring Secure Domain Routers on Cisco IOS XR Software

Secure domain routers (SDRs) are a means of dividing a single physical system into multiple logically separated routers. SDRs are isolated from each other in terms of their resources, performance, and availability.

For complete descriptions of the SDR commands listed in this module, see Related Documents, on page 24.

Table 2: Feature History for Configuring Secure Domain Routers on Cisco IOS XR Software

Release	Modification
Release 3.3.0	This feature was introduced. Support included distributed route processor cards (DRPs) and DRP pairs, and SDR-specific software package activation.
Release 3.5.0	DSC migration functionality was improved.
Release 3.5.2	DSC migration was removed.
Release 3.6.3	Support for an SDR with DRPs within a single rack was added.
Release 3.9.0	Support was added for an SDR with DRPs on different racks.

This module contains the following topics:

Prerequisites for Working with Secure Domain Routers

Before configuring SDRs, the following conditions must be met:

Initial Setup

- The router must be running the Cisco IOS XR software , including a designated shelf controller (DSC).
- The root-system username and password must be assigned as part of the initial configuration.
- For more information on booting a router and performing initial configuration, see *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*.

Required Cards for Each SDR

- Additional route processor (RP) pair, DRP or DRP pair must be installed in each line card (LC) chassis to manage each SDR in the system.
- For additional information on DRPs, refer to *Cisco CRS-1 Carrier Routing System 16-Slot Line Card Chassis System Description*. For instructions on installing DRPs, see *Installing the Cisco CRS-1 Carrier Routing System 16-Slot Line Card Chassis*.

Task ID Requirements

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Software Version Requirements

- Cisco IOS XR Software Releases 2.0, 3.0, and 3.2 support only one owner SDR. Multiple (non-owner) SDRs are not supported in these releases. The owner SDR cannot be added or removed from the configuration.
- Multiple SDRs, including non-owner SDRs, are supported on Cisco IOS XR Software Release 3.3.0 or higher.

Maximum SDR Configurations

- A maximum of eight SDRs are supported, including one owner SDR and up to seven non-owner SDRs.

Information About Configuring Secure Domain Routers

Review the sections in this module before configuring secure domain routers.

What Is a Secure Domain Router?

Cisco routers running the Cisco IOS XR software can be partitioned into multiple independent routers known as Secure Domain Routers (SDRs). An user defined SDR is termed as named-SDR.

SDRs are a means of dividing a single physical system into multiple logically separated routers. The SDRs are spawned as Virtual Machines (VMs). Each SDR performs routing functions similar to a physical router, but they share resources with the rest of the system. For example, the software image, configurations, protocols, and routing tables are unique to a particular SDR. Other system functions, including chassis-control and switch fabric, are shared with the rest of the system.

Owner SDR and Administration Configuration Mode

The *owner SDR* is created at system startup and cannot be removed. This owner SDR performs system-wide functions, including the creation of additional *non-owner* SDRs. You cannot create the owner SDR because it always exists, nor can you completely remove the owner SDR because it is necessary to manage the router. By default, all nodes in the system belong to the owner SDR.

The owner SDR also provides access to the administration EXEC and administration configuration modes. Only users with root-system privileges can access the administration modes by logging in to the primary route processor (RP) for the owner SDR (called the *designated shelf controller*, or DSC).

Administration modes are used for the following purposes:

- Create and remove additional non-owner SDRs.
- Assign nodes to the non-owner SDRs.
- View the configured SDRs in the system.
- View and manage system-wide resources and logs.

Note Administration modes cannot be used to configure the features within a non-owner SDR, or view the router configuration for a non-owner SDR. After the SDR is created, users must log into the non-owner SDR directly to change the local configuration and manage the SDR.

Related Topics

SDR Access Privileges, on page 6
Non-Owner SDRs, on page 5

Non-Owner SDRs

To create a new non-owner SDR, the root-system user enters administration configuration mode, defines a new SDR name, and assigns a set of cards to that SDR. Only a user with root-system privileges can access the commands in administration configuration mode. Therefore, users without root-system privileges cannot create SDRs or assign cards to the SDRs.

After a non-owner SDR is created, the users configured on the non-owner SDR can log in and manage the router. The configuration for each non-owner SDR is separate from the owner SDR and can be accessed only by logging in to the non-owner SDR.

Note For information regarding support for non-owner SDRs in Cisco IOS XR software releases before release 3.9.0, see *Related Topics*.

Related Topics

SDR Access Privileges, on page 6
Prerequisites for Working with Secure Domain Routers, on page 3

SDR Access Privileges

Each SDR in a router has a separate AAA configuration that defines usernames, passwords, and associated privileges.

- Only users with root-system privileges can access the administration EXEC and administration configuration modes.
- Users with root-lr privileges can access only the non-owner SDR in which that username was created.
- Users with other access privileges can access features according to their assigned privileges for a specific SDR.

For more information about AAA policies, see the *Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*.

Related Topics

Root-System Users, on page 6
root-lr Users, on page 7
Other SDR Users, on page 7

Root-System Users

Users with root-system privileges have access to system-wide features and resources, including the ability to create and remove secure domain routers. The root-system user is created during the initial boot and configuration of the router.

The root-system user has the following privileges:

- Access to administration EXEC and administration configuration commands.
- Ability to create and delete non-owner SDRs.
- Ability to assign nodes (RPs, distributed route processors [DRPs], and line cards) to SDRs.
- Ability to create other users with similar or lower privileges.
- Complete authority over the chassis.
- Ability to log in to non-owner SDRs using admin plane authentication. Admin plane authentication allows the root-system user to log in to a non-owner SDR regardless of the configuration set by the root-lr user.
- Ability to install and activate software packages for all SDRs or for a specific SDR .
- Ability to view the following administration (admin) plane events (owner SDR logging system only):
  - Software installation operations and events.
  - System card boot operations, such as card booting notifications and errors, heartbeat-missed notifications, and card reloads.
  - Card alphanumeric display changes.
  - Environment monitoring events and alarms.
  - Fabric control events.
  - Upgrade progress information.

Related Topics

Configuring a Username and Password for a Non-Owner SDR, on page 18

root-lr Users

Users with root-lr privileges can log in to an SDR only and perform configuration tasks that are specific to that SDR. The root-lr group has the following privileges:

- Ability to configure interfaces and protocols.
- Ability to create other users with similar or lower privileges on the SDR.
- Ability to view the resources assigned to their particular SDR.

The following restrictions apply to root-lr users:

- Users with root-lr privileges cannot enter administration EXEC or configuration modes.
- Users with root-lr privileges cannot create or remove SDRs.
- Users with root-lr privileges cannot add or remove nodes from an SDR.
- Users with root-lr privileges cannot create root-system users.
- The highest privilege a non-owner SDR user can have is root-lr.

Other SDR Users

Additional usernames and passwords can be created by the root-system or root-lr users to provide more restricted access to the configuration and management capabilities of the owner SDR or non-owner SDRs.

Designated Secure Domain Router Shelf Controller (DSDRSC)

In a router running Cisco IOS XR software, one RP is assigned the role of DSC. The DSC provides system-wide administration and control capability, including access to the administration EXEC and administration configuration modes. For more information on DSCs, refer to *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*.

In each SDR, similar administration and control capabilities are provided by the designated secure domain router system controller (DSDRSC). Each SDR must include a DSDRSC to operate, and you must assign an RP or DRP to act as the DSDRSC.

Note In the owner SDR, the DSC also provides DSDRSC functionality.

DSCs and DSDRSCs

Designated Shelf Controller (DSC)

The primary and standby DSC is always an RP pair. By default, the DSC is also the DSDRSC for the owner SDR. The owner DSDRSCs cannot be removed from the SDR configuration, or assigned to a non-owner SDR.

For information on DSC assignment and initial router configuration, refer to *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*.

Using a DRP or DRP Pair as the DSDRSC

Cisco Systems recommends the use of DRPs as the DSDRSC in non-owner SDRs. An SDR without an RP must designate a DRP or DRP as the potential DSDRSC.

To create a DRP DSDRSC in a non-owner SDR, you must configure a DRP or DRP pair as the primary node for that SDR. The following guidelines apply:

- Although a single DRP can be used as the DSDRSC, we recommend the use of a redundant DRP pair.
- To create a DRP pair and configure it as the DSDRSC, complete the instructions in Creating SDRs , on page 11.
- DRPs cannot be used as the DSC in the owner SDR. Only RPs can be used as the DSC in the owner SDR.
- DRPs cannot be assigned as the DSDRSC if an RP is present in the SDR. To assign a DRP as the DSDRSC, you must first remove any RPs from the SDR configuration, and then add the DRP or DRP pair as the primary node. After the DRP is assigned as the DSDRSC, the RPs can be added to the SDR. For more information, see *Related Topics*.

Note DRPs can also be used to provide additional processing capacity. For additional information on DRPs, see *Cisco CRS-1 Carrier Routing System 16-Slot Line Card Chassis System Description*. For instructions on installing DRPs, see *Installing the Cisco CRS-1 Carrier Routing System 16-Slot Line Card Chassis*. For information on using DRPs for additional processing capacity, see the *Process Placement on Cisco IOS XR Software* module in *System Management Configuration Guide for Cisco CRS Routers*.

Related Topics

Creating SDRs , on page 11
How to Configure Secure Domain Routers, on page 11
Adding Nodes to a Non-Owner SDR, on page 14
Adding Nodes to an SDR, on page 14
Removing Nodes and SDRs, on page 16
Removing Nodes from an SDR , on page 16
Removing an SDR , on page 18
Configuring a Username and Password for a Non-Owner SDR, on page 18
Disabling Remote Login for SDRs, on page 21

Using an RP Pair as the DSDRSC

RP pairs can also be used as the DSDRSC in non-owner SDRs.

- Single RPs cannot be used as the DSDRSC.
- Redundant RPs are installed in slots RP0 and RP1 of each line card chassis.
- To assign an RP pair as the DSDRSC, complete the instructions in How to Configure Secure Domain Routers, on page 11.

Note Although an RP pair can be used as the DSDRSC in non-owner SDRs, we recommend the use of a redundant DRP pair.

Removing a DSDRSC Configuration

There are two ways to remove a DSDRSC from an SDR:

- First remove all other nodes from the SDR configuration, and then remove the DSDRSC node. You cannot remove the DSDRSC node when other nodes are in the SDR configuration.

- Remove the entire SDR. Removing an SDR name deletes the SDR and moves all nodes back to the owner SDR inventory.

Related Topics

Removing Nodes and SDRs, on page 16

Default Configuration for New Non-Owner SDRs

By default, the configuration of a new SDR is blank. The first configuration step after creating an SDR is to log in to the new non-owner SDR using admin plane authentication and create a username and password. You can then log out of the SDR and log back in using the new username and password.

Note When logged in to a non-owner SDR using admin plane authentication, the admin configuration is displayed. However, admin plane authentication should be only used to configure a username and password for the non-owner SDR. To perform additional configuration tasks, log in with the username for the non-owner SDR.

Default Software Profile for SDRs

When a new non-owner SDR is created, the nodes assigned to that SDR are activated with the default software package profile. The default software profile is defined by the last install operation that did not specify an SDR.

To view the default software profile, use the **show install active summary** command in administration EXEC mode. Any new nodes that are configured to become a part of an SDR will boot with the default software profile listed in the output of this command.

```
RP/0/RP0/CPU0:router# show install active summary

Wed Dec 24 01:47:02.076 PST
  Active Packages:
    disk1:hfr-infra-test-3.8.0.25I
    disk1:hfr-fpd-3.8.0.25I
    disk1:hfr-doc-3.8.0.25I
    disk1:hfr-diags-3.8.0.25I
    disk1:hfr-mgbl-3.8.0.25I
    disk1:hfr-mcast-3.8.0.25I
    disk1:hfr-mpls-3.8.0.25I
    disk1:comp-hfr-mini-3.8.0.25I
```

Note For detailed instructions to add and activate software packages, see the *Managing Cisco IOS XR Software Packages* module of *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*. See also the *Software Package Management Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*.

Related Topics

Configuring a Username and Password for a Non-Owner SDR, on page 18

High Availability Implications

The sections in this module describe various high availability implications.

Fault Isolation

Because the CPU and memory of an SDR are not shared with other SDRs, configuration problems that cause out-of-resources conditions in one SDR do not affect other SDRs.

Rebooting an SDR

Each non-owner SDR can be rebooted independently of the other SDRs in the system. If you reboot the owner SDR, however, then all non-owner SDRs in the system automatically reboot, because the non-owner SDRs rely on the owner SDR for basic chassis management functionality.

Note The DSDRSC of the owner SDR is also the DSC of the entire system.

DSDRSC Redundancy

To achieve full redundancy, each SDR must be assigned two cards: one to act as the primary DSDRSC and one RP or DRP to act as a standby DSDRSC.

We recommend the use of DRP pairs as DSDRSC for all non-owner SDRs the system.

Cisco IOS XR Software Package Management

Software packages are added to the DSC of the system from administration EXEC mode. Once added, a package can be activated for all SDRs in the system or for a specific SDR . For detailed instructions regarding software package management, see the *Upgrading and Managing Cisco IOS XR Software* module of *System Management Configuration Guide for Cisco CRS Routers*. See also the *Software Package Management Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*.

Note SDR-specific activation is supported for specific packages and upgrades, such as optional packages and SMUs. Packages that do not support SDR-specific activation can only be activated for all SDRs in the system.

- To access **install** commands, you must be a member of the root-system user group with access to the administration EXEC mode.
- Most **show install** commands can be used in the EXEC mode of an SDR to view the details of the active packages for that SDR.

Related Topics

Default Configuration for New Non-Owner SDRs, on page 9

Restrictions For SDR Creation and Configuration

The following restrictions apply to SDR creation and configuration:

- DRPs are supported for the DSDRSC.
- We recommend the configuration of DRP pairs as the DSDRSC for all non-owner SDRs, as described in Using a DRP or DRP Pair as the DSDRSC, on page 7.
- Single RPs are not supported for the DSDRSC. RPs must be installed and configured in redundant pairs.
- Admin plane events are displayed only on the non-owner SDR.
- Some admin plane debug events are not displayed on the owner SDR. For example, a non-owner card cannot send debug events to the DSC, which limits the debugging of administration processes to the non-owner SDR.

How to Configure Secure Domain Routers

To create an SDR, configure an SDR name and then add nodes to the configuration. At least one node in each SDR must be explicitly configured as the DSDRSC. After the SDR is created, you can add or remove additional nodes and create a username and password for the SDR.

Creating SDRs

To create a non-owner SDR, create an SDR name, add a DSDRSC, and then add additional nodes to the configuration. After the SDR is created, you can create a username and password for the SDR to allow additional configuration.

Note The Cisco CRS-1 supports a maximum of eight SDRs, including one owner SDR and up to seven non-owner SDRs.

The 4-slot line card chassis does not support the creation of multiple SDRs.

Before you begin

The procedures in this section can be performed only on a router that is already running Cisco IOS XR software. For instructions to boot a router and perform the initial configuration, see *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*. When a router is booted, the owner SDR is automatically created, and cannot be removed. This also includes instructions to create the owner SDR username and password.

SUMMARY STEPS

1. **admin**
2. **configure**
3. **pairing** *pair-name*
4. **location** *partially-qualified-nodeid partially-qualified-nodeid*
5. **exit**
6. **sdr** *sdr-name*
7. Do one of the following:
   - **pair** *pair-name* **primary**
   - **location** *partially-qualified-nodeid* **primary**
8. Do one of the following:

- **location** *partially-qualified-nodeid*
- **location** *pair-name*

9. Repeat Step 8, on page 13 as needed to add nodes to an SDR.
10. **exit**
11. Repeat Step 3, on page 12 through Step 10, on page 13 through as needed.
12. **commit**
13. Create a username and password for the new SDR.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 3	**pairing** *pair-name* **Example:** `RP/0/RP0/CPU0:router(admin-config)# pairing drp1`	(Optional) Enter DRP pairing configuration mode. If the DRP name does not exist, the DRP pair is created when you add nodes, as described in the following step. • *pair-name* can be between 1 and 32 alphanumeric characters. The characters '_' or '-' are also allowed. All other characters are invalid. DRP pairs are used as the DSDRSC for a non-owner SDR. **Note** Although a single DRP can be used as the DSDRSC in a non-owner SDR, Cisco systems recommends that two redundant DRPs be installed and assigned to the SDR. **Note** DRPs can also be added to an SDR to provide additional processing capacity. See *Related Topics* for more information on DRP installation and configuration.
Step 4	**location** *partially-qualified-nodeid* *partially-qualified-nodeid* **Example:** `RP/0/RP0/CPU0:router(admin-config-pairing:drp1)# location 0/3/* 0/4/*`	(Optional) Specifies the location of the DRPs in a DRP pair. The *partially-qualified-nodeid* argument is entered in the *rack/slot/** notation. Node IDs are always specified at the slot level, so the wildcard (*) is used to specify the CPU.
Step 5	**exit** **Example:**	(Optional) Exits the DRP pairing configuration mode and returns to Administration configuration mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(admin-config-pairing:drp1)# exit`	Complete this step only if you created a DRP pair.
Step 6	**sdr** *sdr-name* **Example:** `RP/0/RP0/CPU0:router(admin-config)# sdr rname`	Enters the SDR configuration sub-mode for the specified SDR. • If this SDR does not yet exist, it is created when you add a node, as described in step 7. • If this SDR existed previously, you can add additional slots as described in step 7 and step 8. • Only alphanumeric characters, "-", and "_" are valid characters to include in the *sdr-name* argument.

Step 7	Do one of the following:
	• **pair** *pair-name* **primary**
	• **location** *partially-qualified-nodeid* **primary**
	Example:
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# pair drp1 primary`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# location 0/0/* primary`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# location 0/RP*/* primary`
		Specifies a DSDRSC for the non-owner SDR. You can assign a redundant DRP pair, an RP pair, or a single DRP as the DSDRSC. You cannot assign a single RP as the DSDRSC. Every SDR must contain a DSDRSC.
		• We recommend the use of DRP pairs as the DSDRSC for all non-owner SDRs.
		• The **primary** keyword configures the RPs, DRP pair, or DRP as the DSDRSC. If the **primary** keyword is not used, the node is assigned to the SDR, but it is not be the DSDRSC.
		• If an RP is already assigned to the SDR, it must be removed before a DRP or DRP pair can be assigned as the DSDRSC.

Step 8	Do one of the following:
	• **location** *partially-qualified-nodeid*
	• **location** *pair-name*
	Example:
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# location 0/0/*`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# location drp1`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# location 0/RP*/*`
		Adds additional nodes, DRP pairs, or RP pairs to the SDR.

Step 9	Repeat Step 8, on page 13 as needed to add nodes to an SDR.	Adds additional nodes to the SDR.
Step 10	**exit** **Example:**	(Optional) Exits the SDR configuration submode and returns to Administration configuration mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router (admin-config-sdr:rname)# exit`	**Note** Complete this step only if you need to create additional SDRs.
Step 11	Repeat Step 3, on page 12 through Step 10, on page 13 through as needed.	Creates additional SDRs.
Step 12	**commit**	
Step 13	Create a username and password for the new SDR.	

Related Topics

Additional References, on page 24
Removing Nodes and SDRs, on page 16
Configuring a Username and Password for a Non-Owner SDR, on page 18
DSCs and DSDRSCs, on page 7
Designated Shelf Controller (DSC), on page 7
Using a DRP or DRP Pair as the DSDRSC, on page 7
Using an RP Pair as the DSDRSC , on page 8

Adding Nodes to a Non-Owner SDR

When adding nodes to an existing non-owner SDR, the following rules apply:

- By default, all nodes in a new system belong to the owner SDR. When a node is assigned to a non-owner SDR, the node is removed from the owner SDR inventory and added to the non-owner SDR.
- When a node is removed from a non-owner SDR, it is automatically returned to the owner SDR inventory.
- To add a node that already belongs to another non-owner SDR, you must first remove the node from the other SDR, and then reassign it to the new SDR.
- You cannot assign the DSC or standby DSC to a non-owner SDR. The DSC and standby DSC can cannot be removed and assigned to a non-owner SDR.
- Note the following points about DSDRSC support:
  - DRPs and DRP pairs are supported.
  - RPs can only be added in redundant pairs.

Related Topics

Removing Nodes and SDRs, on page 16

Adding Nodes to an SDR

This task explains how add nodes to an SDR.

SUMMARY STEPS

1. **admin**
2. **configure**
3. **sdr** *sdr-name*
4. Do one of the following:

- **location** *partially-qualified-nodeid*
- **location** *pair-name*

5. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 3	**sdr** *sdr-name* **Example:** `RP/0/RP0/CPU0:router(admin-config)# sdr rname`	Enters the SDR configuration submode for the specified SDR. • *sdr-name* is the name assigned to the SDR.

Step 4	Do one of the following:
	• **location** *partially-qualified-nodeid*
	• **location** *pair-name*
	Example:
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# location 0/0/*`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# location drp1`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# location 0/RP*/*`
		Adds additional nodes, DRP pairs, or RP pairs to an SDR.

Step 5	Use one of the following commands: • **end** • **commit** **Example:** `RP/0/RP0/CPU0:router(admin-config-sdr:rname)# end`	Saves configuration changes. • When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`

	Command or Action	Purpose
	or `RP/0/RP0/CPU0:router(admin-config-sdr:rname)# commit`	• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Removing Nodes and SDRs

When removing a node or an entire SDR, the following rules apply:

- When a node is removed from a non-owner SDR, it is automatically returned to the owner SDR inventory.
- To remove a DSDRSC, first remove the other nodes in the SDR and then remove the DSDRSC. This rule does not apply when the entire SDR is removed.
- If all nodes are removed from a non-owner SDR, the SDR name is also removed.
- To remove all nodes, including the DSDRSC, remove the SDR name. All nodes are returned to the owner SDR inventory.
- You must first remove a node from a non-owner SDR before it can be reassigned to another non-owner SDR.
- To remove a node from the owner SDR inventory, assign the node to an non-owner SDR.
- The owner SDR cannot be removed, and the owner DSDRSC (DSC) cannot be removed.

Removing Nodes from an SDR

This task explains how remove nodes from an SDR.

SUMMARY STEPS

1. **admin**
2. **configure**
3. **sdr** *sdr-name*
4. Do one of the following:
   - **no location** *partially-qualified-nodeid*
   - **no location** *pair-name*
5. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 3	**sdr** *sdr-name* **Example:** `RP/0/RP0/CPU0:router(admin-config)# sdr rname`	Enters the SDR configuration submode for the specified SDR.

Step 4	Do one of the following:
	• **no location** *partially-qualified-nodeid*
	• **no location** *pair-name*
	Example:
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname2)# no location 0/0/*`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname2)# no location drp1`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# no location 0/RP*/*`
		Removes a node, DRP pair, or RP pair from a non-owner SDR.
		• When a node is removed from an SDR, it is automatically added to the owner SDR inventory. This node may now be assigned to a different SDR, as described in Adding Nodes to a Non-Owner SDR, on page 14.
		• Removing all the slots from an SDR deletes that SDR.

Step 5	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# end`
	or
	`RP/0/RP0/CPU0:router(admin-config-sdr:rname)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

	Command or Action	Purpose
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Removing an SDR

This section provides instructions to remove a secure domain router from your router. To remove an SDR, you can either remove all the nodes in the SDR individually or remove the SDR name. This section contains instructions to remove the SDR name and return all nodes to the owner SDR inventory.

Note The owner SDR cannot be removed. Only non-owner SDRs can be removed.

SUMMARY STEPS

1. **admin**
2. **configure**
3. **no sdr** *sdr-name*
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 3	**no sdr** *sdr-name* **Example:** `RP/0/RP0/CPU0:router(admin-config)# no sdr rname`	Removes the specified SDR from the current owner SDR. **Note** All slots belonging to that SDR return to the owner SDR inventory.
Step 4	**commit**	

Configuring a Username and Password for a Non-Owner SDR

After you create an SDR, you can create a username and password on that SDR. When you assign root-lr privileges to that username, the user can administer the non-owner SDR and create additional users if necessary.

Note Only users with root-system privileges can access administration modes to add or remove SDRs. SDR users cannot add or remove SDRs.

To create a username and password for the new non-owner SDR.

1. On the owner SDR, enable admin plane authentication. This allows you to log in to the non-owner SDR and create local usernames and passwords.
2. Log in to the non-owner SDR.
3. Configure a new username and password on the non-owner SDR. Assign the username to the root-lr group to allow the creation of additional usernames on that SDR.
4. To verify the new username, log out and log back in to the non-owner SDR using the new username and password.
5. Provide the username and password to the SDR user.

Complete the following steps to create usernames and passwords on a non-owner SDR.

SUMMARY STEPS

1. Connect a terminal to the console port of the DSC (DSDRSC of the owner SDR).
2. **admin**
3. **configure**
4. **aaa authentication login remote local**
5. **commit**
6. Connect a terminal to the console port of the non-owner SDR DSDRSC.
7. Log in to the non-owner SDR using admin plane authentication.
8. **configure**
9. **username** *username*
10. **secret** *password*
11. **group root-lr**
12. **commit**
13. **exit**
14. Log back in with the SDR administrator username and password you created.
15. Provide the new username and password to the user.
16. Disable admin plane authentication.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Connect a terminal to the console port of the DSC (DSDRSC of the owner SDR).	**Note** If an IP address has not yet been assigned to the Management Ethernet port, you must connect a terminal directly to the console port of the DSC.
Step 2	**admin** **Example:**	Enters administration EXEC mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router# admin`	
Step 3	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 4	**aaa authentication login remote local** **Example:** `RP/0/RP0/CPU0:router(admin-config)# aaa authentication login remote local`	Enables admin plane authentication. • The **remote** keyword specifies a method list that uses remote non-owner SDR for authentication. • The **local** keyword specifies a method list that uses the local username database method for authentication. The local authentication cannot fail because the system always ensures that at least one user is present in the local database, and a rollover cannot happen beyond the local method. **Note** You can also use other methods to enable AAA system accounting, such as TACACS+ or RADIUS servers. See the *Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers* for more information. **Note** When logged in to a non-owner SDR using admin plane authentication, the admin configuration is displayed. However, admin plane authentication should only be used to configure a username and password for the non-owner SDR. To perform additional configuration tasks, log in with the username for the non-owner SDR, as described in the following steps.
Step 5	**commit**	
Step 6	Connect a terminal to the console port of the non-owner SDR DSDRSC.	**Note** A terminal server connection is required for Telnet connections to the console port because an IP address has not yet been assigned to the management Ethernet port.

Step 7	Log in to the non-owner SDR using admin plane authentication.
	Example:
	`Username:xxxx@admin`
	`Password:pppp`
		Logs a root-system user into the SDR using admin plane authentication.
		Note When prompted for the Username, use your username followed by **@admin** .

Step 8	**configure**	

	Command or Action	Purpose
Step 9	**username** *username* **Example:** `RP/0/RP0/CPU0:router(config)# username user1`	Defines an SDR username and enters username configuration mode. The *username* argument can be only one word. Spaces and quotation marks are not allowed.
Step 10	**secret** *password* **Example:** `RP/0/RP0/CPU0:router(config-un)# secret 5 XXXX`	Defines a password for the user.
Step 11	**group root-lr** **Example:** `RP/0/RP0/CPU0:router(config-un)# group root-lr`	Adds the user to the predefined root-lr group. **Note** Only users with root-system authority or root-lr authority may use this option.
Step 12	**commit**	
Step 13	**exit** **Example:** `RP/0/RP0/CPU0:router# exit`	Closes the active terminal session and log off the router.

Step 14	Log back in with the SDR administrator username and password you created.
	Example:
	`Press RETURN to get started.`
	`Username:xxxx`
	`Password:ppppp`
		Logs back in with the SDR administrator username and password you created. This username is used to configure the secure domain router and create other users with fewer privileges.
		• This step verifies proper SDR administrator username and password configuration.
		• After you create the SDR username and password, you need to provide the SDR username and password to the operators who will use that SDR.

Step 15	Provide the new username and password to the user.	—
Step 16	Disable admin plane authentication.	See *Related Topics* for more information.

Related Topics

Disabling Remote Login for SDRs, on page 21

Disabling Remote Login for SDRs

When you disable admin plane authentication, the admin username cannot be used to log in to non-owner SDRs. Only local SDR usernames can be used to log into the SDR.

SUMMARY STEPS

1. **admin**
2. **configure**
3. **no aaa authentication login remote local**

4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 3	**no aaa authentication login remote local** **Example:** `RP/0/RP0/CPU0:router(admin-config)# no aaa authentication login remote local`	Disables remote login.
Step 4	**commit**	

Configuration Examples for Secure Domain Routers

Creating a New SDR: Example

The following example shows how to create a new SDR:

```
admin
  configure
    pairing drp1
      location 0/3/* 0/4/*
      exit
    sdr rname2
      pair pair1 primary
      location 0/0/*
      end
```

Adding Nodes to an SDR: Example

The following example shows how to add nodes to an SDR:

```
admin
  configure
    sdr rname2
      location 0/0/*
      end
```

Removing Notes from an SDR: Example

The following example shows how to remove nodes from an SDR:

```
admin
  configure
    sdr rname2
      no location 0/0/*
      end
```

Removing an SDR from the Router: Example

The following example shows how to remove an SDR from the router:

```
admin
  configure
    no sdr rname2
    end
```

Configuring a Username and Password for a Non-Owner SDR: Example

The following example shows how to connect to the DSC of the owner SDR:

```
admin
  configure
    aaa authentication login remote local
    end
```

To continue, connect a terminal to the console port of the non-owner SDR DSDRSC.

```
Username:xxxx@admin
  Password:xxxx
  configure
   username user1
    secret 5 XXXX
    group root-lr
    end
   exit

  Press RETURN to get started.
  Username:user1
  Password:xxxxx
```

Disabling Remote Login for SDRs: Example

The following example shows how to disable remote login for an SDR:

```
admin
  configure
    no aaa authentication login remote local
```

```
end
```

Additional References

The following sections provide references related to SDR configuration.

Related Documents

Related Topic	Document Title
SDR command reference	*Secure Domain Router Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*
DRP pairing command reference	*Distributed Route Processor Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*
Initial system bootup and configuration information for a router using the Cisco IOS XR software	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*
DRP description and requirements	*Cisco CRS-1 Carrier Routing System 16-Slot Line Card Chassis System Description*
Instructions to install DRP and DRP PLIM cards	*Installing the Cisco CRS-1 Carrier Routing System 16-Slot Line Card Chassis*
Cisco IOS XR master command reference	*Cisco IOS XR Commands Master List for the Cisco CRS Router*
Information about user groups and task IDs	*Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*
Cisco IOS XR interface configuration commands	*Interface and Hardware Component Command Reference for Cisco CRS Routers*
Information about configuring interfaces and other components on the Cisco CRS-1 from a remote Craft Works Interface (CWI) client management application	*Cisco Craft Works Interface User Guide*
Information about AAA policies, including instructions to create and modify users and username access privileges	*Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

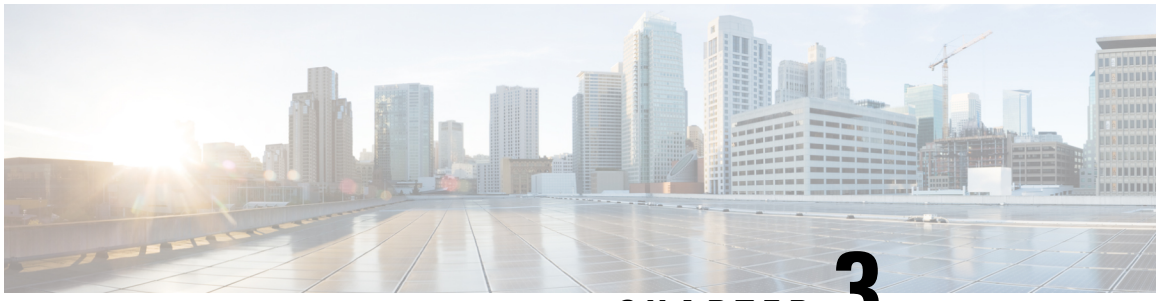
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 3

Upgrading and Managing Cisco IOS XR Software

Cisco IOS XR software is divided into software packages so that you can select which features run on your router. This module describes the concepts and tasks necessary to add feature packages, upgrade the active set of packages, roll back to a previously active set of packages, and perform other related package management tasks.

For complete descriptions of the commands listed in this module, see Related Documents, on page 81. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 3: Feature History for Upgrading and Managing Cisco IOS XR Software

Release	Modification
Release 3.4.0	Support was added for installation operations in SDR EXEC mode. Three new software PIEs were added: • Field-programmable devices (FPD) • Documentation • IPSec Module was moved to *System Management Configuration Guide for Cisco CRS Routers* from *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*.
Release 3.6.0	Support was added for installing PIE files from tar files. Support was added for labels to identify rollback points.

Release	Modification
Release 3.7.0	Support was added for activating software packages by specifying the operation ID of the **install add** command operation that added the packages. Simultaneous activation of multiple SMUs was streamlined to minimize disruption, with fewer cases requiring reloads. The **install verify packages** command was enhanced so that it checks for corruptions of installation state files and MBI image files as well as software files. Support was added for the following commands: • **install boot-options** • **clear boot-options** • **show boot-options**
Release 3.8.0	Support was added for the **pause sw-change** keywords in the following commands: • **install activate** • **install add** • **install deactivate** • **install rollback**
Release 4.0.0	A procedure to upgrade software from Cisco IOS XR Release 3.x was introduced. See Upgrading to Cisco IOS XR Software Release 4.0, on page 66. Support for installation commands was removed from EXEC mode. The ability to install software on a specific SDR was removed.

This module contains the following topics:

Overview of Cisco IOS XR Software Packages

Cisco IOS XR software is divided into software packages so that you can select which features run on your router. Each package contains the components to perform a specific set of router functions, such as routing, security, or modular services card (MSC) support. Bundles are groups of packages that can be downloaded as a set. For example, Cisco IOS XR Unicast Routing Core Bundle (known as *mini*) provides the main packages for use on every router.

Adding a package to the router does not affect the operation of the router—it only copies the package files to a local storage device on the router, known as the *boot device* (such as the internal flash disk0:). To make the package functional on the router, you must activate it for one or more cards.

To upgrade a package, you activate a newer version of the package. When the automatic compatibility checks have been passed, the new version is activated, and the old version is deactivated.

Note Activating a software maintenance upgrade (SMU) does not cause any earlier SMUs or the package to which the SMU applies to be automatically deactivated.

Note If an interface on a router does not have a configuration and is brought up by performing no-shut operation, then upon router reload, the interface state changes to **admin-shutdown** automatically.

To downgrade a package, you activate an older version of the package. When the automatic compatibility checks have been passed, the older version is activated, and the newer version is deactivated.

Caution Do not perform any install operations when the router is reloading.

Note For more information on the features and components included in each package, refer to the release notes.

Package Installation Envelopes

Package Installation Envelopes (PIEs) are nonbootable files that contain a single package or a set of packages (called a *composite package* or *bundle*). Because the files are nonbootable, they are used to add software package files to a running router.

PIE files have a `pie` extension. When a PIE file contains software for a specific bug fix, it is called a *software maintenance upgrade* (SMU).

Note Files with the `vm` extension are bootable installation files used only to replace all current Cisco IOS XR software. These files are installed from ROM Monitor mode, which causes significant router downtime. Cisco Systems recommends installing or upgrading software packages only using PIE files as described in this document. For more information on `vm` files, see *ROM Monitor Configuration Guide for Cisco CRS Routers*.

Summary of Cisco IOS XR Software Packages

Every router includes a basic set of required packages contained in the Cisco IOS XR Unicast Routing Core Bundle. Additional optional packages can be added and activated on the router to provide specific features.

Packages in the Cisco IOS XR Unicast Routing Core Bundle

The packages contained in the Cisco IOS XR Unicast Routing Core Bundle are as follows:

- Operating system (OS) and minimum boot image (MBI)—Kernel, file system, memory management, and other slow changing core components.
- Base—Interface manager, system database, checkpoint services, configuration management, other slow-changing components.
- Infra—Resource management: rack, fabric, secure domain router (SDR).
- Routing—RIB, BGP, ISIS, OSPF, EIGRP, RIP, RPL, and other routing protocols.
- Forwarding—FIB, ARP, QoS, ACL, and other components.
- LC—MSC drivers.

The filename for this bundle is: `hfr-mini.pie-`*`version`*.

Refer to the release notes for additional information on the specific features provided by each package.

Software Maintenance Upgrades

A software maintenance upgrade (SMU) is a PIE file that contains fixes for a specific defect. A composite SMU is a PIE file that contains SMUs for more than one package. SMUs are added and activated using the same procedures as other PIE files. SMUs are created to respond to immediate issues and do not include new features. Typically, SMUs do not have a large impact on router operations. SMU versions are synchronized to the package major, minor, and maintenance versions they upgrade.

The affect of an SMU depends on its type:

- Process Restart SMU—Causes a process or group of processes to restart on activation.
- Reload SMU—Causes a parallel reload (of RPs and line cards).

SMUs are not an alternative to maintenance releases. They provide quick resolution of immediate issues. All bugs fixed by SMUs are integrated into the maintenance releases. For information on available SMUs, contact Cisco Technical Support, as described in *Obtaining Technical Assistance* in the monthly What's New in Cisco Product Documentation.

Note Activating a software maintenance upgrade (SMU) does not cause any earlier SMUs, or the package to which the SMU applies, to be automatically deactivated.

Related Topics

Updating Software Images Without a Router Reload

PIE Filenames and Version Numbers

PIE filenames have two formats: one for composite-package PIEs (bundles) and one for single-package PIEs. A *composite-package file* is a PIE file that contains multiple packages.

Note Hyphens in the filename are part of the filename.

Table 4: PIE Filenames, on page 31 shows the filenames for available PIE types.

Table 4: PIE Filenames

Software Delivery Type	Filename	Example
Composite (Bundle) PIE	*platform-composite_name*-p.pie-*major.minor.maintenance*	`hfr-mini-p.pie-4.0.0`
Single package PIE	*platform-package_type.*-p.pie-*major.minor.maintenance*	`hfr-mgbl-p.pie-3.7.0`
Composite SMU	*platform*-p.*composite_name.ddts*.pie	`hfr-p-4.0.0.16C.CSCee10001.pie`

Filename Component Description

The filename components for all packages are described in Table 5: Composite- and Single-Package Filename Components, on page 31.

Table 5: Composite- and Single-Package Filename Components

Component	Description
platform	Identifies the platform for which the software package is designed. • The platform designation is “hfr.”
composite_name	Identifies a specific composite package. • The only composite PIE file at this time is named “mini” and includes all packages described in the Cisco IOS XR Unicast Routing Core Bundle.
package_type	Identifies the type of package the file supports (*package_type* applies only to single-package PIEs). Package types include: • mcast—Multicast package • mgbl—Manageability package • mpls—MPLS package • k9sec—Security package • diags—Diagnostics package • fpd—Field-programmable device package • cgn—Carrier Grade NAT package • doc—Documentation package
major	Identifies the major release of this package. • A major release occurs when there is a major architectural change to the product (for example, a major new capability is introduced). • All packages operating on the router must be at the same major release level. • A major release is the least frequent release and may require a router reboot.

Component	Description

minor	Identifies the minor release of this package.
	• A minor release contains one or more of the following:
	  • New features
	  • Bug fixes
	• The minor release version does not have to be identical for all software packages operating on the router, but the operating packages must be certified by Cisco as compatible with each other.
	• A minor release may require a router reboot.

maintenance	Identifies the maintenance release of this package. • A maintenance release contains a collection of bug fixes for a package. • The maintenance release version does not have to be identical for all software packages operating on the router, but the major and minor versions of the maintenance release must match those of the package being updated. • A maintenance release does not usually require a router reboot.
ddts	SMUs only. Identifies a DDTS[1] number that describes the problem this SMU addresses. DDTS is the method used to track known bugs and the resolutions or workarounds for those issues.
p	In Cisco IOS XR Software Release 4.0, the software packages were reorganized into functionally well-defined and independently-releasable packages that support the hardware with the PPC architecture. These reorganized packages are identified by the *-p* in the filename. These packages are not compatible with packages released prior to Release 4.0. When upgrading to Release 4.0 or above, special upgrade instructions must be followed.
px	Identifies images that are compatible with hardware that uses the x86 architecture. Starting with Cisco IOS XR Release 4.2, -px releases replace the -p releases.

[1] distributed defect tracking system

Copying the PIE File to a Local Storage Device or Network Server

To add an optional package or upgrade or downgrade a package, you must copy the appropriate PIE file to a local storage device or to a network file server to which the router has access.

If you need to store PIE files on the router, we recommended storing PIE files on the hard disk. Flash disk0: serves as the boot device for packages that have been added or activated on the system. Flash disk1: is used as a backup for disk0:.

Tip Before copying PIE files to a local storage device, use the **dir** command to check to see if the required PIE files are already on the device.

Information About Package Management

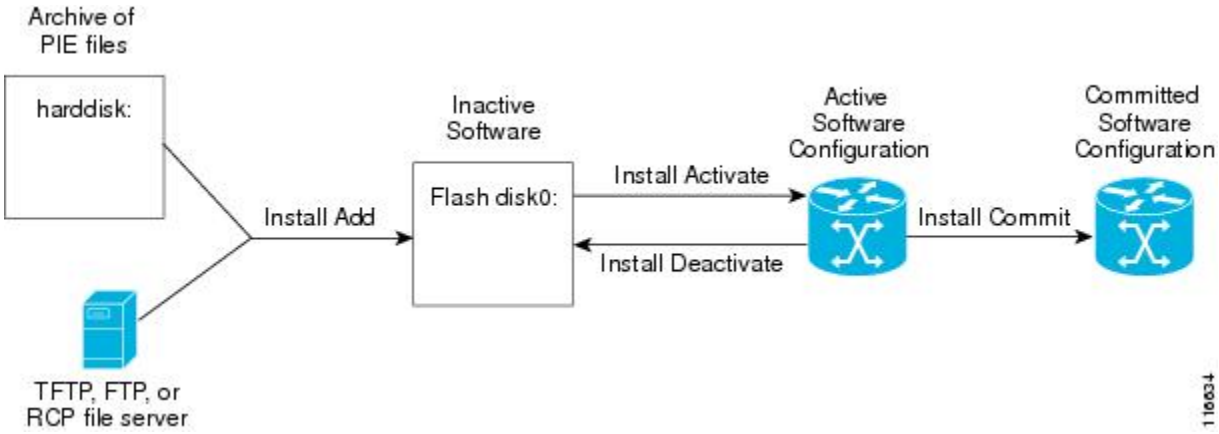
Summary of Package Management

The general procedure for adding optional packages, upgrading a package or package set, or downgrading packages on the router is as follows:

1. Copy the package file or files to a local storage device or file server.
2. Add the package or packages on the router using the command **install add**.
3. Activate the package or packages on the router using the **install activate** command.
4. Commit the current set of packages using the **install commit** command.

Figure 1: Process to Add, Activate, and Commit Cisco IOS XR Software Packages, on page 33 illustrates key steps in the package management process.

Figure 1: Process to Add, Activate, and Commit Cisco IOS XR Software Packages



Adding Packages

Use the **install add** command to unpack the package software files from a PIE file and copy them to the boot device (usually disk0:) of your router.

From administration EXEC mode, the package software files are added to the boot device of the designated secure domain router system controller (DSDRSC) for all SDRs on the router, as well as all active and standby Route Processors (RPs), DRPs and fabric shelf controllers (SCs) installed on the router.

Note The disk that holds the unpacked software files is also known as the *boot device*. By default, flash disk0: is used as the boot device. To use an alternate storage device, such as flash disk1:, see the *Router Recovery with ROM Monitor* module of *ROM Monitor Configuration Guide for Cisco CRS Routers*. Remember that all RPs in a system must use the same boot device. If the boot device on the primary RP is flash disk0:, then the standby RP or DRP must also have a flash disk0:.

Verifying Package Details

Before you activate a package on the router, you can verify the type of upgrade that is required for the package and whether the package requires a router reload or not. Use the **show install package** *pie* **detail** command in admin mode.

```
RP/0/RP0/CPU0:router(admin)# show install package disk0:hfr-px-4.x.x.04I.CSCuc66088-0.0.4.i
 detail

Mon Nov 19 09:44:24.036 UTC
disk0:hfr-px-4.x.x.04I.CSCuc66088-0.0.4.i
    asr9k-px-4.x.x.04I.CSCuc66088 V0.0.4.i[SMU]  User specified bundle
        iosxr-infra-hfr-px1-4.x.x.04I.CSCuc66088.pi.pie.
     [composite package]
     [root package, grouped contents]
    Vendor : Cisco Systems
    Desc   : User specified bundle iosxr-infra-hfr-px1-4.x.x.04I.CSCuc66088.pi.pie.
    Build  : Built on Fri Nov  9 11:00:11 UTC 2012
   Source : By iox-bld27 in /scratch1/SMU_BLD_WS/ci-431_206626_CSCuc66088_121109102249 for
 pie
    Card(s): RP, CRS-RP-X86, CRS8-RP-x86, CRS16-RP-x86, ASR9001-RP, RP-STARSCREAM,
NP24-4x10GE,
           NP24-40x1GE, NP40-40x1GE, NP40-4x10GE, NP40-8x10GE, NP40-2_20_COMBO, NP80-8x10GE,

             NP80-16x10GE, NP200-24x10GE, NP200-36x10GE, NP200-2x100GE, NP200-1x100GE,
NP200-5x40GE,
             NP200-8x10GE, NP200-MOD-SMEM, NP200-MOD-LMEM, ASR9001-LC, A9K-SIP-700,
A9K-SIP-500, A9K-SIP-AVSM
    Restart information:
      Default:
        parallel impacted processes restart
    Size Compressed/Uncompressed: 1744KB/1830KB (95%)
    Components in package disk0:asr9k-px-4.x.x.04I.CSCuc66088-0.0.4.i, package
hfr-px-4.x.x.04I.CSCuc66088:
    disk0:iosxr-infra-4.x.x.04I.CSCuc66088-0.0.4.i
        iosxr-infra-4.x.x.04I.CSCuc66088 V0.0.4.i[SMU]  IOS-XR Infra Package Definition
        Vendor : Cisco Systems
        Desc   : IOS-XR Infra Package Definition
        Build  : Built on Fri Nov  9 11:00:10 UTC 2012
       Source : By iox-bld27 in /scratch1/SMU_BLD_WS/ci-431_206626_CSCuc66088_121109102249
 for pie
        Card(s): RP, CRS-RP-X86, CRS8-RP-x86, CRS16-RP-x86, ASR9001-RP, RP-STARSCREAM,
NP24-4x10GE,
                 NP24-40x1GE, NP40-40x1GE, NP40-4x10GE, NP40-8x10GE, NP40-2_20_COMBO,
NP80-8x10GE,
                 NP80-16x10GE, NP200-24x10GE, NP200-36x10GE, NP200-2x100GE, NP200-1x100GE,

                 NP200-5x40GE, NP200-8x10GE, NP200-MOD-SMEM, NP200-MOD-LMEM, ASR9001-LC,
                 A9K-SIP-700, A9K-SIP-500, A9K-SIP-AVSM

        Size Compressed/Uncompressed: 1744KB/1830KB (95%)
        Components in package disk0:iosxr-infra-4.x.x.04I.CSCuc66088-0.0.4.i,
            package  iosxr-infra-4.x.x.04I.CSCuc66088:
            platforms-spa-chopper V[ci-4x-bugfix/8]  This component contains Platform
Independent
            Chopper SPA Code.
            iosxr-infra-4.x.x.04I.CSCuc66088-package V[Default]  Manifest information for
package
            iosxr-infra-4.x.x.04I.CSCuc66088
            iosxr-infra-4.x.x.04I.CSCuc66088-package-compatibility V[Default]
            Package Compatibility information for package iosxr-infra-4.x.x.04I.CSCuc66088
```

Activating Packages

Software packages remain inactive until activated with the **install activate** command.

After a package has been added to the router , use the **install activate** command to activate the package or SMUs for all valid cards. Information within the package is used to verify compatibility with the target cards and with the other active software. Actual activation is performed only after the package compatibility and application programming interface (API) compatibility checks have been passed.

Activating a Package on the Router

To activate a package on your router, use the **install activate** command in administration EXEC mode. The **install activate** command also activates the package on all administration plane nodes and resources, including service processors (SPs), fabric SCs, fan controllers, alarm modules, and power modules.

Note To enter administration EXEC mode, you must be logged in to the owner secure domain router (SDR) and have root-system access privileges.

Activating Multiple Packages or SMUs

To install multiple packages or software maintenance upgrades (SMUs) with a single command, use the **install activate** command and either specify up to 16 packages by repeating *device: package* arguments or use wildcard syntax to specify multiple packages. Some SMUs may require a reload. If the operation requires a node reload, the user is prompted before the installation operation occurs.

Related Topics

SMU Installation Combinations

Activating All Packages Added in a Specific Operation

To install all packages that were added in a specific **install add** operation, use the **install activate** command with the **id** *add-id* keyword and argument, specifying the operation ID of the **install add** operation. You can specify up to 16 operations in a single command.

Adding and Activating a Package with a Single Command

To add and activate a package with a single command, use the **install add** command with the **activate** keyword from administration EXEC mode.

Upgrading and Downgrading Packages

To upgrade a package, activate the latest version of the package; the previous version is automatically deactivated. To downgrade a package, activate the previous version of the package; the latest version is automatically deactivated.

Actual activation is performed only after compatibility checks have been passed.

Note

- Activating a software maintenance upgrade (SMU) does not cause previous versions of the SMUs, or the package to which the SMU applies, to be automatically deactivated.
- If you upgrade an ASR 9000 router with low RSP card memory, then the RSP440-TR route-switch processor and Cisco ASR 9000 2nd Generation line card can become inaccessible due to insufficient memory. Power cycling the router may help bring the router back to the up state.

Committing the Active Software Set

When a package is activated on the router , it becomes part of the current running configuration for those SDRs. To make the package activation persistent across reloads, enter the **install commit** command in administration EXEC mode. On startup, the designated secure domain router shelf controller (DSDRSC) of the secure domain router (SDR) loads the committed software set.

Note If the system is restarted before the active software set is saved with the **install commit** command, the previously committed software set is used.

Rolling Back to a Previous Installation Operation

Although the term *commit* sounds final, the Cisco IOS XR software provides the flexibility to roll back the selected package set to previously saved package sets. Each time a package is activated or deactivated, a rollback point is created that defines the package set that is active after the package activation or deactivation. The software also creates a rollback point for the last committed package set. If you find that you prefer a previous package set over the currently active package set, you can use the **install rollback** command to make a previously active package set active again.

Related Topics

Rolling Back to a Previous Software Set, on page 78

Multiple Disks Support during Installations

In installations on platforms where Cisco IOS XR Software is supported, only a single disk is used as an install device; that is, either disk0 or disk1. When multiple packages are installed on a single disk, it results in space constraints. To resolve this space limitation, the disk supported for the install operations has been extended to another disk called the disk1. When installing multiple packages, this feature enables you to choose between disk0 and disk1.

To add packages to a specific disk name, use the **install media** command in the admin configuration mode.

```
RP/0/RSP0/CPU0: router (admin) # install media disk1
```

Restrictions

- Before enabling the addition of disk1 through the **install media** command, the disk mirroring feature should be explicitly disabled. For details regarding disk mirroring, see the Disk Mirroring chapter.
- All single version packages should be installed into one disk; that is, either disk0 or disk1.

- When downgrading to an image that does not support extended disk, the rollback points of the extended disk will not be available on the downgraded image. For example, assume a case where the version1 (V1) image does not support the extended disk functionality and version2 (V2) image supports the functionality. Upgrading from V1(disk0) to V2(disk1), in such a case, makes the rollback points of V1 available on V2. However, when downgrading from V2(disk1) to V1(disk0), the rollback points of V2 will not be available on V1. For more information about the rollback feature and rollback points, see the Upgrading and Managing Software chapter.

Deactivation of fully superseded SMUs

Cisco IOS XR Software will accumulate a set of Software Maintenance Upgrades (SMUs) over time, where an older SMU gets superceded by the latest SMU. For example, if SMU A was initially delivered to you, and subsequently, as a result of a bug resolution, SMU B was delivered, then SMU A becomes the subset of SMU B and SMU A is superceded by SMU B. In this case, SMU A is redundant and can be deactivated to clean up the software package.

To deactivate all the fully superseded SMUs, use the **install deactivate superseded** command in the admin mode.

```
RP/0/RSP0/CPU0: router(admin) # install deactivate superseded
```

To display the details of the SMUs that are superseded, use the **show install superseded** command in the EXEC mode.

```
RP/0/RSP0/CPU0: router # show install superseded
Thu Feb 3 17:37:20.379 UTC
disk0:asr9k-px-4.3.0.CSCud93518-1.0.0 is fully superseded by
disk0:asr9k-px-4.3.0.CSCue23747-1.0.0
```

Support for the Ignore Package Presence Check Option

During any software package upgrade in Cisco IOS XR Software, two versions of the packages get stored, both the previous version and the upgraded version. In Route Switch Processor 2 (RSP2), the disk space is insufficient to hold all packages of these two versions. To address this, a new optional keyword, **ignore-pkg-presence-check**, is added to the **install activate** command, which allows upgrading with lesser number of packages. For example, assume a case where version1 (V1) of the software consists of packages A, B, C, and D, and you want to upgrade to the version2 (V2) with only 3 packages (A, B, and C). The ignore-pkg-presence-check option allows only packages A, B, and C to be upgraded to V2 and deactivates package D of V1. Thus, an explicit deactivation of package D is not required and the user can add package D of V1 after upgrading to V2.

To upgrade software with lesser number of packages, use the **install activate [ignore-pkg-presence-check]** command in the admin mode.

```
RP/0/RSP0/CPU0: router(admin) # install activate [ignore-pkg-presence-check] V2 packages
```

Restrictions

The restrictions for this option are:

- The ignore-pkg-presence-check keyword is supported only with the **install activate** command and is not supported with the **install add activate** command.
- When you upgrade using the ignore-pkg-presence-check option, the deactivation of packages always happens synchronously, using the synchronous keyword in the **install deactivate** command.

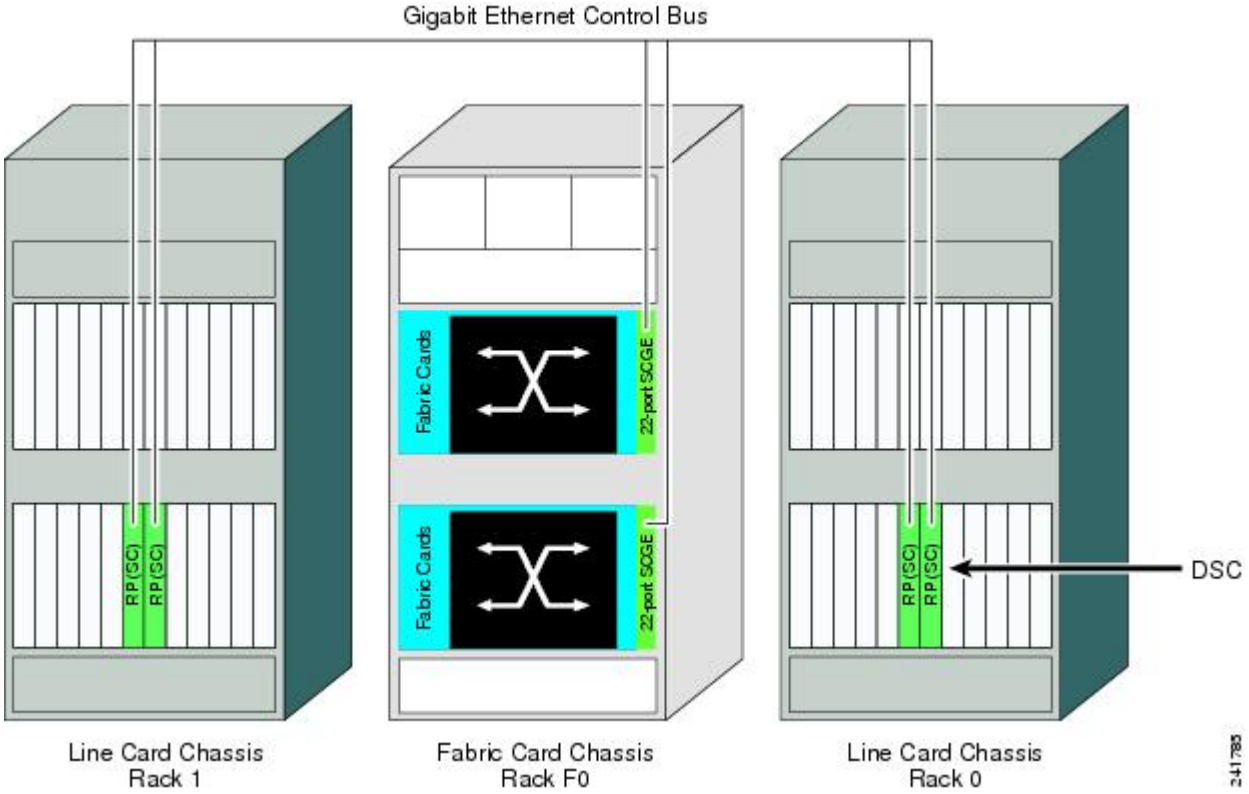
Managing Software Packages in a Multishelf System

Software operations in a multishelf system are the same as in a single-shelf system: software packages are added and activated on the router from administration EXEC mode. The DSC keeps track of software operations for the entire system, while the DSDRSC of each SDR manages the software operations for that specific SDR.

The software packages and related configurations are synchronized throughout a multishelf system by the designated shelf controller (DSC), using the Ethernet control network, as shown in Figure 2: DSC in a CRS-1/M-F1 Multishelf System, on page 38. The DSC maintains an inventory of the packages, versions, and configurations for each node in the system.

Note Whenever a new chassis or node is added to the system, the DSC verifies that the software configuration for that chassis is correct and downloads any required packages and configurations. The active RP in each chassis then distributes and manages the software and configurations for the cards and equipment in that chassis.

Figure 2: DSC in a CRS-1/M-F1 Multishelf System



Default Software Profile for New SDRs

When a new non-owner SDR is created, the nodes assigned to that SDR are activated with the default software profile. The default profile is the active software set for the owner SDR. The owner SDR is the default SDR accessed by logging into the DSC of the system.

To view the default software profile, use the **show install active summary** command in administration EXEC mode. Any new nodes that are configured to become a part of an SDR will boot with the default software profile listed in the output of this command.

```
RP/0/RP0/CPU0:router(admin)# show install active summary

  Default Profile:
    SDRs:
      Owner
      sdr1
    Active Packages:
      disk0:hfr-diags-3.3.0
      disk0:hfr-mgbl-3.3.0
      disk0:hfr-mcast-3.3.0
      disk0:hfr-mpls-3.3.0
      disk0:hfr-k9sec-3.3.0
      disk0:hfr-mini-3.3.0
```

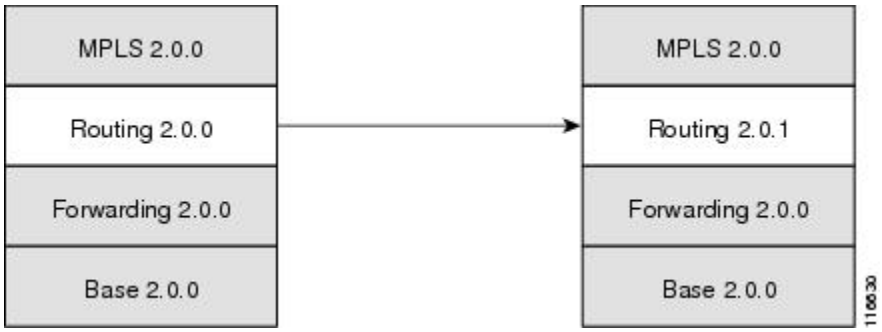
Upgrading Packages

To upgrade a package that is currently active on your SDR, add and activate a newer version of the same package (see Figure 3: Example of a Maintenance Release Package Upgrade, on page 39). The older version of the software package is deactivated automatically. These actions are permitted only after the package compatibility checks and API version compatibility checks have been passed.

Deactivated packages are not removed from the router. To remove inactive package files, use the **install remove** command.

Caution Upgrading or downgrading a software package can cause a process to restart or a new process to start. Use the **test** option to preview the impact of the package activation.

Figure 3: Example of a Maintenance Release Package Upgrade



Related Topics

Deactivating and Removing Cisco IOS XR Software Packages, on page 72

Downgrading Packages

To downgrade a software package, activate an older version on one or more cards for which that package is already active. The newer version of the same software package is deactivated automatically. These actions

are performed only after the package compatibility checks and API version compatibility checks have been passed.

Deactivated packages are not removed from the router. To remove inactive package files, use the **install remove** command. See the *Related Topics* section for links to more information.

Note Downgrading to Cisco IOS XR Software Release 3.7.0 or earlier releases is not supported if you are using a FAT32 flash disk. If you are using a FAT32 flash disk, and you must downgrade, convert the flash disk to FAT16 before downgrading. If you do not convert the flash disk to FAT16 before the downgrade, the disk becomes unreadable and the router does not boot. Converting from FAT32 to FAT16 is a complex procedure. If you need to convert a FAT32 disk to FAT16, contact Cisco technical support for guidance.

Related Topics

Deactivating and Removing Cisco IOS XR Software Packages, on page 72

Impact of Package Version Changes

Each package version change has a different impact on the operation of the router, depending on the type of package and whether the upgrade is for a major, minor, or maintenance release. The following resources can provide more information on the impact of a package version change:

- See *Related Topics* for more information on the typical impact for major, minor, and maintenance releases.
- For specific information about the impact of an upgrade, consult the release notes for the package release, and test the impact of the package activation by adding the test option to the **install activate** command.
- The Cisco IOS XR Software Selector tool also contains information on package version compatibility.

Related Topics

PIE Filenames and Version Numbers, on page 30
Obtaining and Placing Cisco IOS XR Software, on page 45

Impact of Package Activation and Deactivation

Activation or deactivation of a package can have an immediate impact on the system. The system can be affected in the following ways:

- When a new package is activated, any new CLI commands for the package are added to the SDRs impacted by the new software . The router need not be restarted or reloaded.
- When a package is deactivated, the commands associated with the features being deactivated are removed from any SDR impacted by the operation . The commands are no longer available to the user.
- During a software package deactivation, upgrade, or downgrade, any incompatible configurations are removed from the running configuration of any SDR impacted by the operation , and saved to a file. Messages for incompatible configurations are displayed. Incompatible configurations are those configurations that are not supported by the new version of the software package.

Note You must address any issues that result from the revised configuration and reapply the configuration, if necessary.

- New processes may be started.
- Running processes may be stopped or restarted.
- All processes in the cards may be restarted. Restarting processes in the cards is equivalent to a soft reset.
- The cards may reload.
- No impact: no processes in the card may be affected.

Tip When activating and deactivating packages, use the **test** option to test the effects of a command without impacting the running system. After the activation or deactivation process completes, enter the **show install log** command to display the process results.

Delaying the Return of the CLI Prompt

By default, the CLI prompt is returned to the screen before the installation operation is complete, which allows you to enter other commands that are not installation commands. If additional installation requests are attempted before the first operation is complete, they are not run.

To delay the return of the CLI prompt until an installation operation is complete, enter the **install** command with the **synchronous** keyword. For example:

```
install add disk1:/pie-file synchronous
install activate disk0:package synchronous
```

To determine if an **install** command is currently running, enter the **show install request** command.

Displaying Installation Log Information

The install log provides information on the history of the installation operations. Each time an installation operation is run, a number is assigned to that operation.

- Use the **show install log** command to display information about both successful and failed installation operations.
- The **show install log** command with no arguments displays a summary of all installation operations. Specify the *request-id* argument to display information specific to an operation. Use the **detail** or **verbose** keywords to display details for specific operation.
- Use the **detail** or **verbose** keywords to display detailed information, including file changes, nodes that could be reloaded, impact to processes, and impact to Dynamic Link Libraries (DLLs).

Tip By default, the install log stores up to 50 entries. Use the **clear install log-history** command to reset the number of entries to any value from 0 to 255.

Examples

Displaying install log Entries: Example

The following example displays information for the install requests. Use the **verbose** keyword to display detailed information, including files changes, impact to processes, and impact to DLLs.

```
RP/0/RP0/CPU0:router(admin)# show install log verbose

Install operation 1 started by user 'labuser' at 17:48:51 UTC Sat Jun 03 2006.
install add /disk1:hfr-diags-p.pie-PD34-06.06.07
/disk1:hfr-k9sec-p.pie-PD34-06.06.07 /disk1:hfr-mcast-p.pie-PD34-06.06.07
/disk1:hfr-mgbl-p.pie-PD34-06.06.07 /disk1:hfr-mpls-p.pie-PD34-06.06.07
Install operation 1 completed successfully at 17:51:32 UTC Sat Jun 03 2006.

Install logs:
  Install operation 1 'install add /disk1:hfr-diags-p.pie-PD34-06.06.07
  /disk1:hfr-k9sec-p.pie-PD34-06.06.07 /disk1:hfr-mcast-p.pie-PD34-06.06.07
  /disk1:hfr-mgbl-p.pie-PD34-06.06.07 /disk1:hfr-mpls-p.pie-PD34-06.06.07'
  started by user 'labuser' at 17:48:51 UTC Sat Jun 03 2006.
  Info:     The following packages are now available to be activated:
  Info:
  Info:         disk0:hfr-diags-3.4.0.1I
  Info:         disk0:hfr-k9sec-3.4.0.1I
  Info:         disk0:hfr-mcast-3.4.0.1I
  Info:         disk0:hfr-mgbl-3.4.0.1I
  Info:         disk0:hfr-mpls-3.4.0.1I
  Info:
  Install operation 1 completed successfully at 17:51:32 UTC Sat Jun 03 2006.

Install operation 2 started by user 'labuser' at 18:06:32 UTC Sat Jun 03 2006.
install activate disk0:hfr-diags-3.4.0.1I disk0:hfr-k9sec-3.4.0.1I
disk0:hfr-mcast-3.4.0.1I disk0:hfr-mgbl-3.4.0.1I disk0:hfr-mpls-3.4.0.1I
Install operation 2 completed successfully at 18:07:48 UTC Sat Jun 03 2006.

Summary:
  Install method: parallel
  Summary of changes on nodes 0/1/SP, 0/6/SP, 0/SM0/SP, 0/SM1/SP, 0/SM2/SP,
  0/SM3/SP:
    Activated:    hfr-diags-3.4.0.1I
    No processes affected

  Summary of changes on nodes 0/1/CPU0, 0/6/CPU0:
    Activated:    hfr-diags-3.4.0.1I
                  hfr-mcast-3.4.0.1I
                  hfr-mpls-3.4.0.1I
    1 hfr-mpls processes affected (0 updated, 1 added, 0 removed, 0 impacted)
    2 hfr-mcast processes affected (0 updated, 2 added, 0 removed, 0 impacted)

    Summary of changes on nodes 0/RP0/CPU0, 0/RP1/CPU0:
    Activated:    hfr-diags-3.4.0.1I
                  hfr-k9sec-3.4.0.1I
                  hfr-mcast-3.4.0.1I
                  hfr-mgbl-3.4.0.1I
                  hfr-mpls-3.4.0.1I
    6 hfr-mgbl processes affected (0 updated, 6 added, 0 removed, 0 impacted)
    8 hfr-mpls processes affected (0 updated, 8 added, 0 removed, 0 impacted)
    7 hfr-k9sec processes affected (0 updated, 7 added, 0 removed, 0 impacted)
   14 hfr-mcast processes affected (0 updated, 14 added, 0 removed, 0 impacted)

Install logs:
  Install operation 2 'install activate disk0:hfr-diags-3.4.0.1I
  disk0:hfr-k9sec-3.4.0.1I disk0:hfr-mcast-3.4.0.1I disk0:hfr-mgbl-3.4.0.1I
  disk0:hfr-mpls-3.4.0.1I' started by user 'labuser' at 18:06:32 UTC Sat Jun
  03 2006.
  Info:     The changes made to software configurations will not be
  Info:     persistent across system reloads. Use the command 'admin install
  Info:     commit' to make changes persistent.
  Info:     Please verify that the system is consistent following the
  Info:     software change using the following commands:
```

```
  Info:         show system verify
 --More--
```

The following example displays information for a specific install request. Use the **detail** keyword to display additional information, including impact to processes and nodes impacted.

```
RP/0/RP0/CPU0:router(admin)# show install log 2 detail

Install operation 2 started by user 'labuser' at 18:06:32 UTC Sat Jun 03 2006.
install activate disk0:hfr-diags-3.4.0.1I disk0:hfr-k9sec-3.4.0.1I
disk0:hfr-mcast-3.4.0.1I disk0:hfr-mgbl-3.4.0.1I disk0:hfr-mpls-3.4.0.1I
Install operation 2 completed successfully at 18:07:48 UTC Sat Jun 03 2006.

Summary:
  Install method: parallel
  Summary of changes on nodes 0/1/SP, 0/6/SP, 0/SM0/SP, 0/SM1/SP, 0/SM2/SP,
  0/SM3/SP:
    Activated:    hfr-diags-3.4.0.1I
    No processes affected

  Summary of changes on nodes 0/1/CPU0, 0/6/CPU0:
    Activated:    hfr-diags-3.4.0.1I
                  hfr-mcast-3.4.0.1I
                  hfr-mpls-3.4.0.1I
      1 hfr-mpls processes affected (0 updated, 1 added, 0 removed, 0 impacted)
      2 hfr-mcast processes affected (0 updated, 2 added, 0 removed, 0 impacted)

  Summary of changes on nodes 0/RP0/CPU0, 0/RP1/CPU0:
    Activated:    hfr-diags-3.4.0.1I
                  hfr-k9sec-3.4.0.1I
                  hfr-mcast-3.4.0.1I
                  hfr-mgbl-3.4.0.1I
                  hfr-mpls-3.4.0.1I
      6 hfr-mgbl processes affected (0 updated, 6 added, 0 removed, 0 impacted)
      8 hfr-mpls processes affected (0 updated, 8 added, 0 removed, 0 impacted)
      7 hfr-k9sec processes affected (0 updated, 7 added, 0 removed, 0 impacted)
     14 hfr-mcast processes affected (0 updated, 14 added, 0 removed, 0 impacted)

Install logs:
  Install operation 2 'install activate disk0:hfr-diags-3.4.0.1I
  disk0:hfr-k9sec-3.4.0.1I disk0:hfr-mcast-3.4.0.1I disk0:hfr-mgbl-3.4.0.1I
  disk0:hfr-mpls-3.4.0.1I' started by user 'labuser' at 18:06:32 UTC Sat Jun
  03 2006.
  Info:     The changes made to software configurations will not be
  Info:     persistent across system reloads. Use the command 'admin install
  Info:     commit' to make changes persistent.
  Info:     Please verify that the system is consistent following the
  Info:     software change using the following commands:
  Info:         show system verify
  Info:         install verify packages
  Install operation 2 completed successfully at 18:07:48 UTC Sat Jun 03 2006.
```

Package Management Procedures

Note Review the concepts about package management before performing the tasks described in this module.

Related Topics

Information About Package Management, on page 33

Activation and Deactivation Prerequisites

These prerequisites must be met for a package to be activated or deactivated:

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- Verify that all cards are installed and operating properly. For example, do not activate or deactivate packages while cards are booting, while cards are being upgraded or replaced, or when you anticipate an automatic switchover activity.
- If a ROM Monitor upgrade is required for the software package, the upgrade must be completed before the package is activated. For ROM Monitor upgrade information and procedures, see *ROM Monitor Configuration Guide for Cisco CRS Routers*.
- Check the sanity of the configuration file system and recover from any internal inconsistencies by using the **cfs check** command.

```
RP/0/RP0/CPU0:router# cfs check

Tue Sep 20 07:22:03.374 DST

Creating any missing directories in Configuration File system...OK
Initializing Configuration Version Manager...OK
Syncing commit database with running configuration...OK
```

- Clear any inconsistency alarms and remove any failed configurations using the **clear configuration inconsistency** command.

  An inconsistency alarm is set when there is a failure to restore the configuration; this can occur during router startup, or when a line card, modular services card (MSC), or route processor (RP) card is inserted or removed. If an inconsistency alarm is set, a message similar to the one in this example is displayed:

```
RP/0/0/CPU0:May 26 11:58:40.662 : cfgmgr-rp[130]: %MGBL-CONFIGCLI-3
   BATCH_CONFIG_FAIL : 28 config(s) failed during startup. To view
   failed config(s) use the command - "show configuration failed startup"
```

  When the inconsistency alarm is set, all configuration commit operations fail until the alarm is cleared.

- Although more than one version of a software package can be added to a storage device, only one version of a package can be active for any card.
- Some packages require the activation or deactivation of other packages.
- The package being activated must be compatible with the current active software set.

Activation is performed only after the package compatibility checks and API version compatibility checks have been passed. If a conflict is found, an on-screen error message is displayed.

While a software package is being activated, other requests are not allowed to run on any of the impacted nodes. Package activation is completed when a message similar to this one appears:

```
Install operation 2 completed successfully at 20:30:29 UTC Mon Nov 14 2005.
```

Each CLI install request is assigned a request ID, which can be used later to review the events.

Obtaining and Placing Cisco IOS XR Software

This section contains information to locate the available software packages and to transfer them either to a local storage device or to a network server. When this is done, the package or packages can be added and activated on the router .

There are two primary ways to obtain packages in Cisco IOS XR software:

- Request the software from Cisco on a flash disk that you can insert into the removable flash disk slot (usually flash disk1:). Flash disk1: is optional. When it is installed, flash disk1: can be used to store PIE files, which can then be used to add new software to the boot device (usually flash disk0:).
- Download the Cisco IOS XR software packages to a local storage device of the DSC, such as flash disk1:, or to a remote server, such as a tftp or rcp server.

The boot device is the local disk on the DSC where Cisco IOS XR software is added and activated. PIE files should not be stored on this boot device. The default boot device is disk0:. All PIE files should be stored on flash disk1:.

Transferring Installation Files from a Network File Server to a Local Storage Device

If the Cisco IOS XR software PIE files are located on a remote TFTP, FTP, SFTP, or rcp server, you can copy the files to a local storage device such as disk1:. When the PIE files are located on a local storage device, the software packages can be added and activated on the router from that storage device. Table 6: Download Protocols Supported by Cisco IOS XR Software, on page 45 describes the supported server protocols, and the CLI syntax used copy files from each server type to the local storage device.

Tip Cisco IOS XR software PIE files can also be added to the router boot device directly from the remote server.

Note Consult your system administrator for the location and availability of your network server.

Table 6: Download Protocols Supported by Cisco IOS XR Software

Name	Description
Trivial File Transfer Protocol	TFTP allows files to be transferred from one computer to another over a network, usually without the use of client authentication (for example, username and password). It is a simplified version of FTP. **Note** Some Cisco IOS XR software images may be larger than 32 MB, and the TFTP services provided by some vendors may not support a file this large. If you do not have access to a TFTP server that supports files larger than 32 MB, download the software image using FTP or rcp.
File Transfer Protocol	FTP is part of the TCP/IP protocol stack and requires a username and password.
Remote Copy Protocol	The rcp protocol uses TCP to ensure the reliable delivery of data, and rcp downloads require a usernames.

Name	Description
SSH File Transfer Protocol	SFTP is part of the SSHv2 feature in the Security package and provides for secure file transfers. For more information, see the *System Security Configuration Guide for Cisco CRS Routers*.

The router commands listed in Table 7: Commands for Copying Package Files to the Router, on page 46 show how to copy package files to the router using three types of file transfer protocols.

Table 7: Commands for Copying Package Files to the Router

Server Type	Command and Examples
TFTP	The following command syntax is used: **copy tftp:**// *hostname_or_ipaddress* / *directory-path* / *pie-name* **disk1:** Example: `RP/0/RP0/CPU0:router# copy tftp://10.1.1.1/images/comp-hfr-mini.pie disk1:`
FTP	The following command syntax is used: **copy ftp:**// *username* **:** *password* **@** *hostname_or_ipaddress* / *directory-path* / *pie-name* **disk1:** Example: `RP/0/RP0/CPU0:router# copy ftp://john:secret@10.1.1.1/images/comp-hfr-mini.pie disk1:`
rcp	The following command syntax is used: **copy rcp:**// *username* **@** *hostname_or_ipaddress* / *directory-path* / *pie-name* **disk1:** Example: `RP/0/RP0/CPU0:router# copy rcp://john@10.1.1.1/images/comp-hfr-mini.pie disk1:`

Table 8: Command Variables for Copying and Adding Packages from a Network Server, on page 46 describes the command variables for copying packages from a network server.

Table 8: Command Variables for Copying and Adding Packages from a Network Server

Variable	Description
hostname_or_ipaddress	Host name or IP address of the server that stores the source file.
pie-name	Name of the PIE file (package). See the Overview of Cisco IOS XR Software Packages, on page 28 for descriptions of the available packages.
username	Required for FTP and rcp only and must be a valid username on the FTP or rcp server.

Variable	Description
password	Required for FTP only. If a password is not provided, the networking device accepts anonymous FTP.
directory-path	The specified directory should be a directory under the home directory of the user. In the rcp and FTP examples in Table 7: Commands for Copying Package Files to the Router, on page 46, the file being downloaded is in a subdirectory called "images" in the home directory of the user "john." **Note** For FTP and rcp services, *directory-path* is the directory relative to the *username* home directory. If you want to specify an absolute path for the directory, you must add a "/" following the server address.

When the installation files have been transferred to a network file server or the router, you are ready to activate or upgrade the software.

Note Files with the `vm` extension are bootable installation files used only to replace all current Cisco IOS XR software. These files are installed from ROM monitor mode and cause significant router downtime. We recommend installing or upgrading software packages using PIE files only, as described in this chapter. See *ROM Monitor Configuration Guide for Cisco CRS Routers* for information on installing from `vm` files.

Related Topics

Adding and Activating Packages, on page 55
Overview of Cisco IOS XR Software Packages, on page 28

Preparing for Software Installation Operations

This section includes instructions to prepare for software installation operations.

Note Activation is performed only after the automatic package compatibility and API version compatibility checks have been passed. If a conflict is found, an on-screen error message is displayed.

Before you begin

Before adding or activating Cisco IOS XR software:

- Update the ROM Monitor software, if necessary.
- Determine if a software change is required.
- Verify that the new package is supported on your system. Some software packages require that other packages or package versions be activated, and some packages only support specific cards.
- Review the release notes for important information related to that release and to help determine the package compatibility with your router configuration.
- Verify that the system is stable and prepared for the software changes.

SUMMARY STEPS

1. **admin**
2. **show diag**
3. Update the ROMMON software if necessary.
4. **show install active**
5. **show install pie-info** *device:package* [**brief** | **detail** | **verbose**]
6. **verify packages**
7. **exit**
8. (Optional) **show system verify start**
9. (Optional) **show system verify** [**detail** | **report**]
10. **show clock**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**show diag** **Example:** `RP/0/RP0/CPU0:router(admin)# show diag`	Displays the ROMMON software version for all cards in the system. Verify that the correct ROMMON software version is installed before upgrading a Cisco IOS XR software package. **Note** See *Related Topics* for information regarding the required ROM Monitor (ROMMON) software version.
Step 3	Update the ROMMON software if necessary.	Updates the ROMMON software. For instructions, see *ROM Monitor Configuration Guide for Cisco CRS Routers*.
Step 4	**show install active** **Example:** `RP/0/RP0/CPU0:router(admin)# show install active`	Displays the active software on the router for an SDR or for all SDRs . Use this command to determine what software should be added, upgraded or downgraded on the router, and to compare to the active software report after installation operations are complete. **Note** You can also display the active packages for a specific node, and view results in detailed or summary mode. See the *Software Package Management Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers* for more information.
Step 5	**show install pie-info** *device:package* [**brief** | **detail** | **verbose**] **Example:**	Displays information imbedded in the package. The following keywords provide three levels of information:

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(admin)# show install pie-info disk1:/hfr-mcast-p.pie-3.8.30`	• **brief** (default)—Displays the expiration date of the file, the size, and the installed package name. The expiration date is used for certifying the package. • **detail**—Displays the package components, the compatible cards, the expiration date, file size, and the installed package name. • **verbose**—Displays information from the **detail** display and sub-component information. **Note** Always review the release notes for the software package for important information related to that release and to help determine the package compatibility with your router configuration.
Step 6	**verify packages** **Example:** `RP/0/RP0/CPU0:router(admin)# install verify packages`	Verifies that there are no corrupted software files. The consistency of a previously installed software set is verified against the package file from which it originated. This command can be used as a debugging tool to verify the validity of the files that constitute the packages, to determine if there are any corrupted files. This command also checks for corruptions of installation state files and MBI image files. This command is particularly useful when issued after the activation of a package or upgrading the Cisco IOS XR software to a major release. **Note** The **install verify packages** command can take up to two minutes per package to process.
Step 7	**exit** **Example:** `RP/0/RP0/CPU0:router(admin)# exit`	Exits administration EXEC mode and returns to EXEC mode.
Step 8	(Optional) **show system verify start** **Example:** `RP/0/RP0/CPU0:router# show system verify start`	Starts the system status check.
Step 9	(Optional) **show system verify** [**detail** | **report**] **Example:** `RP/0/RP0/CPU0:router# show system verify`	Displays system status information. A variety of information is displayed including the memory and CPU usage, process status, protocol status, and other status information. Use this information to verify that the system is stable. • **detail**—Displays additional information at the card and processor level, including actual numbers. • **report**—Displays the same information as the default **show system verify** command

	Command or Action	Purpose
		Note Although most of the output should display the status "OK," some processes may show other output, such as "Warning." This does not specifically indicate a problem. Contact your Cisco technical support representative for more information on the output of this command.
Step 10	**show clock** **Example:** `RP/0/RP0/CPU0:router# show clock`	Verifies that the system clock is correct. Software operations use certificates based on router clock times.

Related Topics

Activation and Deactivation Prerequisites, on page 44

Examples

Verifying That the ROM Monitor Version Is Correct: Example

In the following example, the ROM Monitor software version is displayed in the "ROMMON:" field for each card.

Note For instructions to upgrade the ROM Monitor software, see *ROM Monitor Configuration Guide for Cisco CRS Routers*.

```
RP/0/RP0/CPU0:router# admin
RP/0/RP0/CPU0:router(admin)# show diag

CARD 0/1/* : Cisco CRS-1 Series Modular Services Card
  MAIN:  board type 500060
         800-25021-05 rev B0
         dev 079239
         S/N SAD09280BS9
  PCA:   73-7648-08 rev B0
  PID:   CRS-MSC
  VID:   V02
  CLEI:  IPUCAC1BAA
  ECI:   132502

PLIM 0/PL1/* : Cisco Carrier Routing System SPA Interface Processor Card
  MAIN:  board type 580070
         800-23819-03 rev C0
         dev N/A
         S/N SAD094401CR
  PCA:   73-8982-06 rev C0
  PID:   CRS1-SIP-800
  VID:   V01
  CLEI:  COUIAAMCAA
  ECI:   134912
  Interface port config:  0 Ports
```

```
  Optical reach type:  Unknown
  Connector type:  MT-P

NODE 0/1/SP
  Node State : IOS XR RUN
  PLD:   Motherboard: 0x0025, Processor: 0xda13, Power: N/A
  MONLIB: QNXFFS Monlib Version 3.1
  ROMMON: Version 1.49(20080319:195528)  [CRS-1 ROMMON]

NODE 0/1/CPU0
  Node State : IOS XR RUN
  PLD:   Motherboard: 0x0025, Processor: 0xda13, Power: N/A
  MONLIB: QNXFFS Monlib Version 3.0
  ROMMON: Version 1.49(20080319:195807)  [CRS-1 ROMMON]
--More--
```

Displaying the Active Software for All SDRs or for a Specific SDR: Example

The following example displays the active packages for all SDRs in the system. Use this information to determine if a software change is required:

```
RP/0/RP1/CPU0:router(admin)# show install active summary

  Default Profile:
    SDRs:
      Owner
      CE1b
  Active Packages:
      disk0:hfr-diags-3.3.30
      disk0:hfr-mgbl-3.3.30
      disk0:hfr-k9sec-3.3.30
      disk0:comp-hfr-mini-3.3.30
```

The following example displays a summary of active packages for a specific SDR:

```
RP/0/RP1/CPU0:router(admin)# show install active summary sdr owner

  Active Packages:
      disk0:hfr-diags-3.3.30
      disk0:hfr-mgbl-3.3.30
      disk0:hfr-k9sec-3.3.30
      disk0:comp-hfr-mini-3.3.30
```

Displaying Information About the Contents of a PIE File: Example

In the following example, information is displayed about the manageability PIE. This command displays the expiry date of the package, the cards supported by the package, and other details. Use this information to verify the compatibility of the package with your system and other software packages.

Note A software activation is performed only after the automatic package compatibility and API version compatibility checks have been passed. If a conflict is found, an on-screen error message is displayed.

```
RP/0/RP0/CPU0:router(admin)# show install pie-info disk1:/
hfr-mgbl-p.pie-3.8.0 detail

Contents of pie file '/disk1:/hfr-mgbl-p.pie-3.8.0':
    Expiry date       : Jan 19, 2007 02:55:56 UTC
    Uncompressed size : 17892613

    hfr-mgbl-3.8.0
        hfr-mgbl V3.8.0[00]  Manageability Package
        Vendor : Cisco Systems
        Desc   : Manageability Package
        Build  : Built on Wed May 10 08:04:58 UTC 2006
        Source : By edde-bld1 in /vws/aga/production/3.8.0/hfr/workspace for c28
        Card(s): RP, DRP, DRPSC
        Restart information:
          Default:
            parallel impacted processes restart
        Components in package hfr-mgbl-3.8.0, package  hfr-mgbl:
            manageability-cwi V[r33x/2]  Craft Web Interface related binaries ae
            hfr-feature-ipsla V[r33x/1]  IPSLA time stamping feature
            doc-hfr-mgbl V[r33x/2]  Contains the man page documentation for hfrs
 --More--
```

Verifying That There Are No Corrupted Software Files: Example

The following sample output verifies the consistency of the currently active software against the file from which it originated:

```
RP/0/RP0/CPU0:router# install verify packages

  Install operation 2 '(admin) install verify packages' started by user 'admin' via CLI at
 07:35:01 UTC Wed May 14 2008.
  Info: This operation can take up to 2 minutes per package being verified. Please be
patient.
  Info: 0/3/CPU0 [LC] [SDR: Owner]
  Info:   meta-data: [SUCCESS] Verification Successful.
  Info:   /install/hfr-lc-3.8.0.02I: [SUCCESS] Verification Successful.
  Info:   /install/hfr-fwdg-3.8.0.02I: [SUCCESS] Verification Successful.
  Info:   /install/hfr-admin-3.8.0.02I: [SUCCESS] Verification Successful.
  Info:   /install/hfr-base-3.8.0.02I: [SUCCESS] Verification Successful.
  Info:   /install/hfr-os-mbi-3.8.0.02I: [SUCCESS] Verification Successful.
  Info: 0/SM1/SP [SP] [Admin Resource]
  Info:   meta-data: [SUCCESS] Verification Successful.
  Info:   /install/hfr-admin-3.8.0.02I: [SUCCESS] Verification Successful.
  Info:   /install/hfr-base-3.8.0.02I: [SUCCESS] Verification Successful.
  Info:   /install/hfr-os-mbi-3.8.0.02I: [SUCCESS] Verification Successful.
  Info: 0/3/SP [SP] [Admin Resource]
  Info:   meta-data: [SUCCESS] Verification Successful.
  Info:   /install/hfr-admin-3.8.0.02I: [SUCCESS] Verification Successful.
  Info:   /install/hfr-base-3.8.0.02I: [SUCCESS] Verification Successful.
  Info:   /install/hfr-os-mbi-3.8.0.02I: [SUCCESS] Verification Successful.
  Info: 0/RP1/CPU0 [RP] [SDR: Owner]
```

```
Info:   meta-data: [SUCCESS] Verification Successful.
Info:   meta-data: [SUCCESS] Verification Successful.
Info:   /install/hfr-os-mbi-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-base-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-admin-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-fwdg-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-lc-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-rout-3.8.0.02I: [SUCCESS] Verification Successful.
Info: 0/RP0/CPU0 [RP] [SDR: Owner]
Info:   meta-data: [SUCCESS] Verification Successful.
Info:   meta-data: [SUCCESS] Verification Successful.
Info:   /install/hfr-os-mbi-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-base-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-admin-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-fwdg-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-lc-3.8.0.02I: [SUCCESS] Verification Successful.
Info:   /install/hfr-rout-3.8.0.02I: [SUCCESS] Verification Successful.
Info: Verification Summary:
Info: 0/3/CPU0: SUCCESSFUL. No anomalies found.
Info: 0/SM1/SP: SUCCESSFUL. No anomalies found.
Info: 0/3/SP: SUCCESSFUL. No anomalies found.
Info: 0/RP1/CPU0: SUCCESSFUL. No anomalies found.
Info: 0/RP0/CPU0: SUCCESSFUL. No anomalies found.
Info: The system needs no repair.
Install operation 2 completed successfully at 07:46:29 UTC Wed May 14 2008.
```

Verifying the Current System Status: Example

The following example shows how to prepare for system verification:

```
RP/0/RP0/CPU0:router# show system verify start

  Storing initial router status ...
  done.
```

The following example shows output from running the **show system verify** command.

Note Although most of the output should display the status "OK," some processes may show other output, such as "Warning." This does not specifically indicate a problem. Contact your Cisco technical support representative for more information on the output of this command.

```
RP/0/RP0/CPU0:router# show system verify

  Getting current router status ...
  System Verification Report
  ==========================
  - Verifying Memory Usage
  - Verified Memory Usage                                  : [OK]
  - Verifying CPU Usage
  - Verified CPU Usage                                     : [OK]

  - Verifying Blocked Processes
  - Verified Blocked Processes                             : [OK]
  - Verifying Aborted Processes
  - Verified Aborted Processes                             : [OK]
```

```
 - Verifying Crashed Processes
 - Verified Crashed Processes                            : [OK]

 - Verifying LC Status
 - Verified LC Status                                    : [OK]
 - Verifying QNET Status
Unable to get current LC status info
 - Verified QNET Status                                  : [FAIL]

 - Verifying GSP Fabric Status
 - Verified GSP Fabric Status                            : [OK]
 - Verifying GSP Ethernet Status
   gsp WARNING messages for router
   Current set of gsp ping nodes does not match initial set of nodes
 - Verified GSP Ethernet Status                          : [WARNING]

 - Verifying POS interface Status
 - Verified POS interface Status                         : [OK]
 - Verifying TenGigE interface Status
 - Verified TenGigE interface Status                     : [OK]

 - Verifying TCP statistics
 - Verified TCP statistics                               : [OK]
 - Verifying UDP statistics
   tcp_udp_raw WARNING messages for router
   UDP Packets sent has not increased during this period.
 - Verified UDP statistics                               : [WARNING]
 - Verifying RAW statistics
 - Verified RAW statistics                               : [OK]

 - Verifying RIB Status
 - Verified RIB Status                                   : [OK]
 - Verifying CEF Status
 - Verified CEF Status                                   : [OK]
 - Verifying CEF Consistency Status
 - Verified CEF Consistency Status                       : [OK]
 - Verifying BGP Status
 - Verified BGP Status                                   : [OK]
 - Verifying ISIS Status
 - Verified ISIS Status                                  : [OK]
 - Verifying OSPF Status
 - Verified OSPF Status                                  : [OK]

 - Verifying Syslog Messages
 - Verified Syslog Messages                              : [OK]

 System may not be stable. Please look into WARNING messages.
```

Verifying That the System Clock Is Correct: Example

The following example displays the current system clock setting:

```
RP/0/RP0/CPU0:router# show clock

02:14:51.474 PST Wed Jan 28 2009
```

Adding and Activating Packages

The procedure in this section describes how to upgrade or add Cisco IOS XR software PIE files that are stored on a local storage device, such as a flash disk, or on a remote TFTP, FTP, SFTP, or rcp server. The PIE software file can include any of the following:

- The Cisco IOS XR Unicast Routing Core Bundle (six packages in one composite PIE file)
- Any of the optional packages (one package per PIE file)
- Software maintenance upgrades (SMUs)

When you need to add and activate two or more of the preceding package types, you should add and activate them in the order listed above.

Note When adding and activating two or more packages, optional packages can be activated together. Also, if the operation is a reload, multiple packages can be activated together. For example, five reload SMUs can be activated together or the Cisco IOS XR Unicast Routing Core Bundle plus the SMUs and optional packages can be activated together.

For a description of the software management process, see the *Related Topics* section.

These instructions are also used to downgrade software packages.

Note By default, installation operations are performed asynchronously: the CLI prompt is returned before the operation is complete, allowing the operator to continue work while the installation is completed in the background. Use the **synchronous** keyword at the end of install commands to delay the return of the CLI prompt until an installation operation is complete. See the *Related Topics* section for more information.

Before you begin

Before upgrading or adding packages, verify that these prerequisites have been met:

- Verify that the ROMMON version is correct. For instructions on upgrading ROM Monitor, see *ROM Monitor Configuration Guide for Cisco CRS Routers*.
- All packages to be upgraded or added are present on a local storage device (for example a flash disk), or a network file server.
- Prerequisites for the activation of packages are met as described in the Prerequisites section.
- Complete the procedures described in the Preparing for Software Installation Operations, on page 47 section.

Note To use the automatic FPD upgrade feature, the **fpd auto-upgrade** command must be enabled in administration configuration mode.

SUMMARY STEPS

1. Connect to the console port and log in.
2. (Optional) **dir** *flash-disk* **:**

3. **admin**
4. **install add** [**source** *source-path* | **tar**] file [**activate**]
5. (Optional) **show install inactive summary**
6. **install activate** {**id** *add-id* | *device package*} [**test**] [**location** *node-id*] [**pause sw-change**] [**prompt-level** {**all** | **none**}] [**auto-abort-timer** {*time* | **off**}]
7. Repeat Step 4, on page 56 through Step 6, on page 58 until all packages are activated.
8. (Optional) **show install active summary**
9. (Optional) **install verify packages**
10. (Optional) **exit**
11. (Optional) **show system verify start**
12. **admin**
13. (Optional) **install commit**
14. Upgrade the field-programmable device (FPD) software, if necessary.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Connect to the console port and log in.	Establishes a CLI management session with the SDR. Connect to the console port for the active DSC. For more information on console connections, see *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*.
Step 2	(Optional) **dir** *flash-disk* **:** **Example:** `RP/0/RP0/CPU0:router# dir disk1:`	Displays the package files that are available for package upgrades and additions. **Note** Only PIE files can be added and activated using this procedure.
Step 3	Required: **admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode. **Note** Some **show install** commands can be entered in EXEC mode on an SDR.

Step 4	**install add** [**source** *source-path*	**tar**] file [**activate**]
		Example:
		`RP/0/RP0/CPU0:router(admin)# install add disk1:/hfr-mgbl-px.pie-6.0.1`
		or
		`RP/0/RP0/CPU0:router(admin)# install add source tftp://10.1.1.1/images/ hfr-k9sec-p.pie hfr-mpls-p.pie hfr-mcast-p.pie`
		or
		`RP/0/RP0/CPU0:router(admin)# install add`
			Unpacks a PIE file from local storage device or network server and adds the package files to the boot device of the router. The boot device is located on the DSC.
			• If the **source** keyword is used, the *source-path* specifies the directory path that is used for multiple filenames in the same directory.
			• If the **tar** keyword is used, all PIE files contained in the tar file are unpacked.
			The *file* argument can take any of these formats:
			• *device filename*
			• **tftp:**//*hostname_or_ipaddress* /*directory-path* /*filename*

	Command or Action	Purpose

	`ftp://john:secret@10.1.1.1/images/hfr-k9sec-p.pie`
	
	or
	
	`RP/0/RP0/CPU0:router(admin)# install add tar rcp://john@10.1.1.1/images/hfr-iosxr-3.6.0.tar`
		• **ftp://** *username***:***password***@***hostname_or_ipaddress*/*directory-path*/*filename*
		• **rcp://***username***@***hostname_or_ipaddress*/*directory-path*/*filename*
		
		These are descriptions for each of the terms used here:
		• *device*—Name of the local storage device where the PIE file is stored, such as **disk1:**/.
		• *filename*—Name of the PIE file you want to add. If the **tar** keyword is used, the *file* argument is the name of a tar file containing one or more PIE files, or directories containing PIE files.
		• **tftp://**—Unpacks the PIE file from a network server using Trivial File Transfer Protocol.
		• **ftp://**—Unpacks the PIE file from a network server using File Transfer Protocol.
		• **rcp://**—Unpacks the PIE file from a network server using Remote Copy Protocol
		• *hostname_or_ipaddress*—Host name or IP address of the network file server.
		• *directory-path*—Network file server path that leads to the PIE file to be added.
		• *username*—Username of user that has access privileges to the directory in which the PIE file is stored.
		• *password*—Password associated with the username of user that has access privileges to the directory in which the PIE file is stored.
		• **activate**—Automatically activates the software package after it is successfully added.
		
		Note Multiple versions of a software package can be added to the storage device without impacting the running configuration, but only one version of a package can be activated for a card.
		
		Tip The automatic FPD upgrade occurs only when the FPD pie is added and activated together with the install PIE.

Step 5	(Optional) **show install inactive summary** **Example:** `RP/0/RP0/CPU0:router(admin)# show install inactive summary`	Displays the inactive packages on the router. Verify that the package added in the previous step appears in the display.

	Command or Action	Purpose
Step 6	**install activate** {**id** *add-id* | *device package*} [**test**] [**location** *node-id*] [**pause sw-change**] [**prompt-level** {**all** | **none**}] [**auto-abort-timer** {*time* | **off**}] **Example:** `RP/0/RP0/CPU0:router(admin)# install activate disk0:hfr-mini-px-4.3.99`	Activates a package that was added to one or more SDRs . (Skip this step if the package was activated earlier with the **install add** command.) • **id** *add-id*—Specifies the package using the operation ID of the **install add** operation in which you added the package. The operation ID is provided in the output of the **install add** command. You can also use **show install log** to display installation operation IDs. • *device:package*—Specifies the package by name. Replace the *device:package* argument with the name of the boot device and inactive package, which can be displayed as described in the previous step. **Note** Press **?** after a partial package name to display all possible matches available for activation. If there is only one match, press [TAB] to fill in the rest of the package name. • **location** *node-id*—Activates a package for a specific card (node). To display a list of node IDs for the entire system, enter the **show platform** command in administration EXEC mode. A package cannot be activated on a single node unless some version of the package being activated is already active on all nodes. **Note** By default, packages are activated for all cards supported by that package. • **pause sw-change**—Pauses the operation after preparatory checks and before the configuration is locked for the actual activation. This action enables you to hold the operation while you perform configuration changes, and proceed with the activation whenever you choose. This operation is useful, for example, if your workflow involves configuring a router out of the network during software installation and you want to minimize the time that the router is out of the network. Follow onscreen instructions to control the pausing and completion of the operation. • **prompt-level**—Use a prompt-level of **all** to view all stages of the installation process and to specify whether to continue, or not. • **auto-abort-timer**—Specifies an abort timer value, in minutes, which when expired loads the last committed loadpath. The default is 60. The timer is enabled by default. After the installation, if the activated software is working correctly, use the **install**

	Command or Action	Purpose
		commit command to cancel the timer and commit the new loadpath. **Note** The package being activated must be compatible with the currently active software to operate. When an activation is attempted, the system runs an automatic compatibility check to ensure that the package is compatible with the other active software on the router. The activation is permitted only after all compatibility checks have been passed. **Tip** When activating packages, use the **test** option to test the effects of a command without impacting the running system. After the activation process finishes, enter the **show install log** command to display the process results. **Tip** The automatic FPD upgrade occurs only when the FPD pie is added and activated together with the install PIE.
Step 7	Repeat Step 4, on page 56 through Step 6, on page 58 until all packages are activated.	Activates additional packages as required.
Step 8	(Optional) **show install active summary** **Example:** `RP/0/RP0/CPU0:router(admin)# show install active`	Displays all active packages. Use this display to determine if the correct packages are active:
Step 9	(Optional) **install verify packages** **Example:** `RP/0/RP0/CPU0:router(admin)# install verify packages`	Verifies the consistency of a installed software set with the package file from which it originated. This command can be used as a debugging tool to verify the validity of the files that constitute the packages, to determine whether there are any corrupted files. This command also checks for corruptions of installation state files and MBI image files. This command is particularly useful when issued after the activation of a package or upgrading the Cisco IOS XR software to a major release. **Note** The **install verify packages** command can take up to two minutes for each package to process.
Step 10	(Optional) **exit** **Example:** `RP/0/RP0/CPU0:router(admin)# exit`	Exits administration EXEC mode and returns to EXEC mode.

	Command or Action	Purpose
Step 11	(Optional) **show system verify start** **Example:** `RP/0/RP0/CPU0:router# show system verify start`	Starts the system status check.
Step 12	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

Step 13	(Optional) **install commit**
	Example:
	`RP/0/RP0/CPU0:router# dir disk1:`
	`RP/0/RP0/CPU0:router(admin)# install commit`
		Commits the current set of packages on the router so that these packages are used if the router is restarted.
		For more information, see the *Related Topics* section.

Step 14	Upgrade the field-programmable device (FPD) software, if necessary.	Whenever a Cisco IOS XR software image that supports SPAs and SIPs is released, a companion SPA or SIP FPD image is bundled with the Cisco IOS XR software release. Generally, the FPD image is not automatically upgraded. You must manually upgrade the FPD image running on the SPA or SIP when you upgrade the Cisco IOS XR software image. FPD versions must be compatible with the Cisco IOS XR software that is running on the router. **Note** If you have enabled the **fpd auto-upgrade** command and add and activate the FPD PIE together with the software installation PIE, the FPD image is automatically upgraded before the router is rebooted. For information on FPDs, including instructions to upgrade FPD images, see the *Upgrading FPD Cisco IOS XR Software* section.

Related Topics

Obtaining and Placing Cisco IOS XR Software, on page 45
Activation and Deactivation Prerequisites, on page 44
Preparing for Software Installation Operations, on page 47
Information About Package Management, on page 33
Downgrading Packages, on page 39
PIE Filenames and Version Numbers, on page 30
Committing the Active Package Set, on page 64
Prerequisites for ISSU
Restrictions for ISSU
ISSU Software Images

Examples

Adding a Package: Example

The following example shows how to add the contents of a PIE file on disk1: to the boot device. Because the software package is added to the boot device by default, it is not necessary to specify the destination device in the CLI.

Note From Cisco IOS XR Software Release 6.0.1 and later, you must append a forward slash ('/') to the source location (for example, **disk1:/**) of the PIE file in the **install add** command.

```
RP/0/RP0/CPU0:router(admin)# install add disk1:/hfr-mpls-px.pie-6.0.1 synchronous

Install operation 4 'install add /disk1:/hfr-mpls.pie-mpls-px.pie-6.0.1 synchronous' started
 by user
'cisco' at 18:10:18 UTC Sat Apr 08 2009.
Info:     The following package is now available to be activated:
Info:
Info:         disk0:hfr-mpls-px.pie-6.0.1
Info:
Install operation 4 completed successfully at 18:14:11 UTC Sat Apr 08 2009.
```

The following example shows how to add the contents of a PIE file on a TFTP server to the boot device:

```
RP/0/RP0/CPU0:router(admin)# install add tftp://209.165.201.1/
hfr-mpls.pie synchronous

Install operation 4 '(admin) install add /tftp://209.165.201.1/hfr-mpls.pie synchronous'
  started by user 'cisco' at 18:16:18 UTC Thu Jan 03 2009.
Info:     The following package is now available to be activated:
Info:
Info:         disk0:hfr-mpls-3.7.2
Info:
Install operation 4 completed successfully at 18:19:10 UTC Thu Jan 03 2009.
```

Activating a Package: Example

The following example shows the activation of the MPLS package. The package is activated on the boot device disk0:.

```
RP/0/RP0/CPU0:router(admin)# install activate disk0:
hfr-mpls-3.7.2 synchronous

Install operation 15 'install activate disk0:hfr-mpls-3.7.2 synchronous'
started by user 'lab' at 19:15:33 UTC Sat Apr 08 2009.
Info:     The changes made to software configurations will not be persistent
Info:     across system reloads. Use the command 'admin install commit' to make
Info:     changes persistent.
Info:     Please verify that the system is consistent following the software
Info:     change using the following commands:
Info:         show system verify
```

```
Info:         install verify packages
Install operation 5 completed successfully at 19:16:18 UTC Sat Apr 08 2009.
```

Activating a Package by Specifying an Operation ID: Example

The following example shows the activation of the MPLS package using the operation ID of the **install add** operation that added the package:

```
RP/0/RP0/CPU0:router(admin)# install activate id 4

Install operation 5 '(admin) install activate id 4' started by user 'lab' via
CLI at 18:20:17 UTC Thu Jan 03 2009.
Info:     This operation will activate the following package:
Info:         disk0:hfr-mpls-3.7.2
Info:     Install Method: Parallel Process Restart
The install operation will continue asynchronously.
Info:     The changes made to software configurations will not be persistent
Info:     across system reloads. Use the command '(admin) install commit' to
Info:     make changes persistent.
Info:     Please verify that the system is consistent following the software
Info:     change using the following commands:
Info:         show system verify
Info:         install verify packages
Install operation 5 completed successfully at 18:21:30 UTC Thu Jan 03 2009.
```

Adding and Activating a Package from an FTP File Server with One Command: Example

To add and activate a package with a single command, enter the **install add** command with the **activate** keyword. In the following example, the Manageability PIE located on disk1: is verified, unpacked, and added to the boot device disk0. Because this operation is performed in administration EXEC mode, the package is activated for all SDRs in the system.

```
RP/0/RP0/CPU0:router(admin)# install add disk1:/
hfr-mgbl-px.pie-6.0.1 activate

Install operation 4 'install add /disk1:/hfr-mgbl-px.pie-6.0.1 activate' started
by user 'cisco' at 07:58:56 UTC Wed Mar 01 2009.
The install operation will continue asynchronously.
:router(admin)#Part 1 of 2 (add software): Started
Info:     The following package is now available to be activated:
Info:
Info:         disk0:hfr-mgbl-px.pie-6.0.1
Info:
Part 1 of 2 (add software): Completed successfully
Part 2 of 2 (activate software): Started
Info:     The changes made to software configurations will not be
  persistent across system reloads. Use the command 'admin install
Info:     commit' to make changes persistent.
Info:     Please verify that the system is consistent following
  the software change using the following commands:
Info:         show system verify
Info:         install verify packages
Part 2 of 2 (activate software): Completed successfully
Part 1 of 2 (add software): Completed successfully
Part 2 of 2 (activate software): Completed successfully
```

```
Install operation 4 completed successfully at 08:00:24 UTC Wed Mar 01 2009.
```

Displaying the Active Packages: Example

The following example displays a summary of the active packages on a router. Because this operation is performed in administration EXEC mode, the active packages for all SDRs are displayed.

```
RP/0/RP0/CPU0:router(admin)# show install active summary

Active Packages:
 disk0:hfr-mpls-3.7.0
 disk0:hfr-mgbl-3.7.0
 disk0:hfr-mcast-3.7.0
 disk0:hfr-k9sec-3.7.0
 disk0:hfr-fpd-3.7.0
 disk0:hfr-doc-3.7.0
 disk0:hfr-diags-3.7.0
 disk0:comp-hfr-mini-3.7.0
```

You can also display the active packages for a specific SDR, or for a specific node. Enter the **show install active** command in EXEC mode, or use the **sdr** keyword in administration EXEC mode, as shown in the following example:

```
RP/0/RP0/CPU0:router(admin)# show install active sdr owner

  Secure Domain Router: Owner

  Node 0/1/CPU0 [LC] [SDR: Owner]
    Boot Device: bootflash:
    Boot Image: /disk0/hfr-os-mbi-3.7.0/mbihfr-lc.vm
    Committed Packages:
      disk0:hfr-fpd-3.7.0
      disk0:hfr-diags-3.7.0
      disk0:hfr-mcast-3.7.0
      disk0:hfr-mpls-3.7.0
      disk0:comp-hfr-mini-3.7.0

  Node 0/4/CPU0 [DRP] [SDR: Owner]
    Boot Device: disk0:
    Boot Image: /disk0/hfr-os-mbi-3.7.0/mbihfr-drp.vm
    Committed Packages:
      disk0:hfr-fpd-3.7.0
      disk0:hfr-doc-3.7.0
      disk0:hfr-diags-3.7.0
      disk0:hfr-mgbl-3.7.0
      disk0:hfr-mcast-3.7.0
      disk0:hfr-mpls-3.7.0
      disk0:hfr-k9sec-3.7.0
      disk0:comp-hfr-mini-3.7.0

  Node 0/4/CPU1 [DRP] [SDR: Owner]
    Boot Device: disk0:
    Boot Image: /disk0/hfr-os-mbi-3.7.0/mbihfr-drp.vm
    Committed Packages:
      disk0:hfr-fpd-3.7.0
      disk0:hfr-doc-3.7.0
      disk0:hfr-diags-3.7.0
      disk0:hfr-mgbl-3.7.0
```

```
    disk0:hfr-mcast-3.7.0
    disk0:hfr-mpls-3.7.0
    disk0:hfr-k9sec-3.7.0
    disk0:comp-hfr-mini-3.7.0

Node 0/RP0/CPU0 [RP] [SDR: Owner]
  Boot Device: disk0:
  Boot Image: /disk0/hfr-os-mbi-3.7.0/mbihfr-rp.vm
  Committed Packages:
    disk0:hfr-fpd-3.7.0
    disk0:hfr-doc-3.7.0
    disk0:hfr-diags-3.7.0
    disk0:hfr-mgbl-3.7.0
    disk0:hfr-mcast-3.7.0
    disk0:hfr-mpls-3.7.0
    disk0:hfr-k9sec-3.7.0
    disk0:comp-hfr-mini-3.7.0
```

Committing the Active Package Set

When a package is activated, it becomes part of the current running configuration. To make the package activation persistent across system-wide reloads, enter the **install commit** command. On startup, the DSDRSC of the SDR loads this committed software set. If the system is reloaded before the current active software is committed with the **install commit** command, the previously committed software set is used.

Note If the SDR reloads and the committed SDR software is incompatible with the current software running on the rest of the system, the committed software of the SDR will not be used and the current running SDR software is used.

If the system is reloaded before the current active software is committed with the **install commit** command, the previously committed software set is used.

Tip Before committing a package set, verify that the SDR is operating correctly and is forwarding packets as expected.

SUMMARY STEPS

1. **admin**
2. **install commit**
3. **show install committed** [**detail** | **summary** | **verbose**] [**location** *node-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

	Command or Action	Purpose
Step 2	Required: **install commit** **Example:** `RP/0/RP0/CPU0:router(admin)# install commit`	Commits the current set of packages on the router so that these packages are used if the router is restarted.
Step 3	**show install committed** [**detail** | **summary** | **verbose**] [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router(admin)# show install committed`	Displays which packages are committed.

Examples

Committing the Active Package Set: Example

In the following example, the active software packages are committed on all SDRs in the router :

```
RP/0/RP0/CPU0:router(admin)# install commit

Install operation 16 'install commit' started by user 'lab' at 19:18:58 UTC
Sat Apr 08 2009.
Install operation 16 completed successfully at 19:19:01 UTC Sat Apr 08 2009.
```

Displaying the Committed Package Versions: Example

In the following example, the committed packages are shown for the owner SDR:

```
RP/0/RP0/CPU0:router# show install committed

  Secure Domain Router: Owner

    Node 0/1/SP [SP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.3.30/sp/mbihfr-sp.vm
      Committed Packages:
        disk0:hfr-diags-3.3.30
        disk0:comp-hfr-mini-3.3.30

    Node 0/1/CPU0 [LC] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.3.30/lc/mbihfr-lc.vm
      Committed Packages:
        disk0:hfr-diags-3.3.30
        disk0:comp-hfr-mini-3.3.30

    Node 0/6/SP [SP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.3.30/sp/mbihfr-sp.vm
      Committed Packages:
        disk0:hfr-diags-3.3.30
        disk0:comp-hfr-mini-3.3.30

    Node 0/6/CPU0 [LC] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.3.30/lc/mbihfr-lc.vm
      Committed Packages:
```

```
--More--
```

As with the **show install active** command, the **show install committed** command may display a composite package that represents all packages in the Cisco IOS XR Unicast Routing Core Bundle.

Upgrading to Cisco IOS XR Software Release 4.0

In Cisco IOS XR Software Release 4.0, the software packages were reorganized into functionally well-defined and independently-releasable packages. For this reason, when you upgrade from a software release prior to Release 4.0, you must perform the following procedure in order to synchronize all of the software packages according to the reorganized structure. General information regarding the the addition and activation of software packages is not covered in this procedure.

The main difference between the standard upgrade procedure and the procedure required to upgrade from Release 3.x to 4.x is that the later requires the addition of one additional software package, known as the *upgrade package* (hfr-upgrade-p.pie).

Before you begin

Before performing this procedure, see the adding and activating software package procedures described in this module.

SUMMARY STEPS

1. **admin**
2. **install add tftp:**// *hostname_or_ipaddress* / *directory-path* / *mandatory-bundle-pie*
3. **install add tftp:**// *hostname_or_ipaddress* / *directory-path* / **hfr-upgrade-p.pie**
4. **install activate** *device*:*mandatory-bundle-pie device*:*upgrade-package*
5. **install deactivate** *device*:*upgrade-package*
6. (Optional) **install commit**
7. **install remove** *device*:*upgrade-package*

DETAILED STEPS

	Command or Action	Purpose
Step 1	Required: **admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**install add tftp:**// *hostname_or_ipaddress* / *directory-path* / *mandatory-bundle-pie* **Example:** `RP/0/RP0/CPU0:router(admin)# install add tftp://10.1.1.1/auto/tftpboot/usr/400/hfr-mini-p.pie`	Unpacks the mandatory bundle PIE file from a network server and adds the package file to the boot device of the router. **Note** Refer to the standard procedure to add and activate packages to see other options of PIE file locations and a description of the various arguments for the **install add** command.

	Command or Action	Purpose
Step 3	**install add tftp://** *hostname_or_ipaddress* **/** *directory-path* **/ hfr-upgrade-p.pie** **Example:** `RP/0/RP0/CPU0:router(admin)# install add tftp://10.1.1.1/auto/tftpboot/usr/400/hfr-upgrade-p.pie`	Unpacks the upgrade PIE file from a network server and adds the package file to the boot device of the router.
Step 4	**install activate** *device***:***mandatory-bundle-pie device***:***upgrade-package* **Example:** `RP/0/RP0/CPU0:router(admin)# install activate disk0:hfr-mini-p-4.0.0 disk0:hfr-upgrade-p-4.0.0`	Activates the package that was added to the router together with the upgrade package. **Note** The bundle of mandatory packages and the upgrade bundle are activated together to perform the successful upgrade from release 3.x to 4.x.
Step 5	**install deactivate** *device***:***upgrade-package* **Example:** `RP/0/RP0/CPU0:router(admin)# install deactivate disk0:hfr-upgrade-p-4.0.0`	Deactivates the upgrade package on the router. For specific information regarding the deactivation and removal of software packages, refer to the general procedure.
Step 6	(Optional) **install commit** **Example:** `RP/0/RP0/CPU0:router(admin)# install commit`	Commits the current set of packages so that these packages are used if the router is restarted. Packages can be removed only if the deactivation operation is committed.
Step 7	Required: **install remove** *device***:***upgrade-package* **Example:** `RP/0/RP0/CPU0:router(admin)# install remove disk0:hfr-upgrade-p-4.0.0`	Removes the inactive upgrade package.

Example

The following example illustrates the upgrade operation:

```
RP/0/RP0/CPU0:router(admin)# install add /tftp://223.255.254.254/auto/tftpboot/users/user/
hfr-mini-p.pie

Fri Jul  9 03:53:11.052 UTCRP/0/RP1/CPU0:Jul  9 03:53:12.053 :
instdir[235]: %INSTALL-INSTMGR-6-INSTALL_OPERATION_STARTED :
Install operation 4 '(admin) install add
/tftp://223.255.254.254/auto/tftpboot/users/user/hfr-mini-p.pie'
started by user 'lab'
Install operation 4 '(admin) install add
/tftp://223.255.254.254/auto/tftpboot/users/user/hfr-mini-p.pie'
started by user 'lab' via CLI at 03:53:12 UTC Fri Jul 09 2010.
The install operation will continue asynchronously.
RP/0/RP0/CPU0:router(admin)#
Info:    The following package is now available to be activated:
Info:    disk0:hfr-mini-p-4.0.0
Info:    The package can be activated across the entire router.
```

```
Info:    RP/0/RP1/CPU0:Jul  9 04:32:26.152 : instdir[235]:
%INSTALL-INSTMGR-6-INSTALL_OPERATION_COMPLETED_SUCCESSFULLY :
Info:    Install operation 4 completed successfully
Info:    Install operation 4 completed successfully at 04:32:26 UTC Fri Jul 09 2010.
RP/0/RP0/CPU0:router(admin)# install add /tftp://223.255.254.254/auto/tftpboot/users/user/
hfr-mpls-p.pie

Fri Jul  9 05:07:52.237 UTCRP/0/RP1/CPU0:Jul  9 05:07:53.710 : instdir[235]:
%INSTALL-INSTMGR-6-INSTALL_OPERATION_STARTED :
Info:    Install operation 5 '(admin) install add
Info:    /tftp://223.255.254.254/auto/tftpboot/users/user/hfr-mpls-p.pie'
Info:    started by user 'lab'
Info:    Install operation 5 '(admin) install add
Info:    /tftp://223.255.254.254/auto/tftpboot/users/user/hfr-mpls-p.pie'
Info:    started by user 'lab' via CLI at 05:07:53 UTC Fri Jul 09 2010.
Info:    The install operation will continue asynchronously.
RP/0/RP0/CPU0:router(admin)#
Info:    RP/0/RP1/CPU0:Jul  9 05:09:08.854 : instdir[235]:
%INSTALL-INSTMGR-6-INSTALL_OPERATION_COMPLETED_SUCCESSFULLY :
Install operation 5 completed successfully
Info:    The following package is now available to be activated:
Info:    disk0:hfr-mpls-p-4.0.0
Info:    The package can be activated across the entire router.
Info:    Install operation 5 completed successfully at 05:09:08 UTC Fri Jul 09 2010.
RP/0/RP0/CPU0:router# install add /tftp://223.255.254.254/auto/tftpboot/users/user/
hfr-upgrade-p.pie

Fri Jul  9 05:10:31.133 UTCRP/0/RP1/CPU0:Jul  9 05:10:32.156 : instdir[235]:
%INSTALL-INSTMGR-6-INSTALL_OPERATION_STARTED :
Info:    Install operation 6 '(admin) install add
Info:    /tftp://223.255.254.254/auto/tftpboot/users/user/hfr-upgrade-p.pie'
Info:    started by user 'lab'
Info:    Install operation 6 '(admin) install add
Info:    /tftp://223.255.254.254/auto/tftpboot/users/user/hfr-upgrade-p.pie'
Info:    started by user 'lab' via CLI at 05:10:32 UTC Fri Jul 09 2010.
Info:    The install operation will continue asynchronously.
RP/0/RP0/CPU0:router(admin)#RP/0/RP1/CPU0:
   Jul  9 05:11:55.634 : instdir[235]:
%INSTALL-INSTMGR-6-INSTALL_OPERATION_COMPLETED_SUCCESSFULLY :
Info:    Install operation 6 completed successfully
Info:    The following package is now available to be activated:
Info:    disk0:hfr-upgrade-p-4.0.0
Info:    The package can be activated across the entire router.
Info:    Install operation 6 completed successfully at 05:11:55 UTC Fri Jul 09 2010.
RP/0/RP0/CPU0:router(admin)# install activate disk0:hfr-mini-p-4.0.0 disk0:hfr-upgrade-p-4.0.0
 disk0:hfr-mpls-p-4.0.0

Fri Jul  9 05:23:23.150 UTC
Install operation 7 '(admin) install activate disk0:hfr-mini-p-4.0.0
Info:    disk0:hfr-upgrade-p-4.0.0 disk0:hfr-mpls-p-4.0.0'
Info:    started by user 'lab'RP/0/RP1/CPU0:Jul  9 05:23:24.161 : instdir[235]:
%INSTALL-INSTMGR-6-INSTALL_OPERATION_STARTED :
Info:    Install operation 7 '(admin) install activate disk0:hfr-mini-p-4.0.0
Info:    disk0:hfr-upgrade-p-4.0.0 disk0:hfr-mpls-p-4.0.0'
Info:    started by user 'lab' via CLI at 05:23:24 UTC Fri Jul 09 2010.\ 1% complete:
Info:    The operation can still be aborted (ctrl-c for options)
Info:    This operation will reload the following nodes in parallel:
Info:    0/RP1/CPU0 (HRP) (SDR: Owner)
Info:    0/SM0/SP (Fabric-SP) (Admin Resource)Proceed with this install operation (y/n)?
[y]|
Info:    1% complete: The operation can still be aborted (ctrl-c for options)
Info:    Install Method: Parallel Reload/ 1% complete: The operation can still be aborted
(ctrl-c for options)
Info:    The install operation will continue asynchronously.
```

```
RP/0/RP0/CPU0:router(admin)#SP/0/SM0/SP:
   Jul  9 05:36:41.152 : insthelper[62]: %INSTALL-INSTHELPER-6-RELOAD_NODE_INFO :
Info:    As part of install operation 7 this node (0/SM0/SP) will now reload.
Info:    The changes made to software configurations will not be persistent
Info:    across system reloads. Use the command '(admin) install commit' to
Info:    make changes persistent.
Info:    Please verify that the system is consistent following the software
RP/0/RP1/CPU0:Jul  9 05:36:43.962 : instdir[235]:
%INSTALL-INSTMGR-6-INSTALL_OPERATION_COMPLETED_SUCCESSFULLY :
Info:    Install operation 7 completed successfully
Info:    change using the following commands:
Info:    show system verify
Info:    install verify packages
Info:    Install operation 7 completed successfully at 05:36:43 UTC Fri Jul 09 2010.
rebooting .......................Initializing DDR SDRAM...found 4096 MB
Initializing ECC on bank 0Initializing ECC on bank 1
Initializing ECC on bank 2
Initializing ECC on bank 3
Turning off data cache, using DDR for first time
Initializing NVRAM...Testing a portion of DDR SDRAM ...done
Reading ID EEPROMs ............................
Initializing SQUID ...
Initializing PCI ...PCI0 device[1]: Vendor ID 0x10eePCI0 device[1]: Device ID 0x300ePCI1
device[1]:
Device ID 0x1100PCI1 device[1]: Vendor ID 0x1013PCI1 device[2]: Device ID 0x680PCI1 device[2]:

Vendor ID 0x1095PCI1 device[3]: Device ID 0x5618PCI1 device[3]: Vendor ID 0x14e4Configuring
 MPPs ...
Configuring PCMCIA slots ...System Bootstrap, Version 1.53(20090311:225342) [CRS-1 ROMMON],

Copyright (c) 1994-2009 by Cisco Systems, Inc.
Acquiring backplane mastership ... successful
Preparing for fan initialization.............. ready
Setting fan speed to 4000 RPMs  successfulReading backplane EEPROM ...
Released backplane mastership ...Board type is 0x100002 (1048578)
Switch 0 initialized
Switch 0 Port fe1: link up (100Mb Full Duplex Copper)
Enabling watchdogG4(7457-NonSMP-MV64360 Rev 3) platform with 4096 MB of main memory....

CARD_RACK_NUMBER: 0    CARD_SLOT_NUMBER: 1        CPU_INSTANCE: 1
RACK_SERIAL_NUMBER: TBC08052402
MBI Validation starts ... using Control Plane Ethernet.
DEBUG : Driving up signal strength for Intel LXT971
Our MAC address is 0005.9a3e.89da
Interface link changed state to UP.
Interface link state up.
MBI validation sending request.
HIT CTRL-C to abort
MBI validation sending request.
HIT CTRL-C to abort
MBI validation sending request.
HIT CTRL-C to abort
MBI validation sending request.
HIT CTRL-C to abort
MBI validation sending request.
HIT CTRL-C to abort
No MBI confirmation received from dSCboot: booting from
bootflash:disk0/hfr-os-mbi-4.0.0/mbihfr-rp.vm
.................................................................................
################################################################################

Restricted Rights LegendUse, 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 XR Software for the Cisco XR Router, Version 4.0.0 Copyright (c) 2010 by Cisco
Systems, Inc.
Jul 09 05:39:21.334 : Install (Node Preparation): Booting with software activated by previous
 install
operation,errno=2
RP/0/RP1/CPU0Jul  9 05:44:45.941: syslogd_helper: [89]: dsc_event_handler: Got SysMgr dSC
event : 1
RP/0/RP1/CPU0:Jul  9 05:45:11.354 : shelfmgr[306]: %PLATFORM-SHELFMGR-3-POWERDOWN_RESET :
Node 0/2/SP is powered off due to admin power off request ios con0/RP1/CPU0 is now available
Press RETURN to get started.
RP/0/RP1/CPU0:Jul  9 05:45:27.453 : instdir[216]:
%INSTALL-INSTMGR-4-ACTIVE_SOFTWARE_COMMITTED_INFO :
The currently active software is not committed. If the system reboots then the committed
software will be used.
Use 'install commit' to commit the active software. SYSTEM CONFIGURATION IN PROCESS
The startup configuration for this device is presently loading.
This may take a few minutes. You will be notified upon completion.
Please do not attempt to reconfigure the device until this process is complete.
User Access VerificationUsername: labPassword:
RP/0/RP0/CPU0:router# admin
Fri Jul  9 05:45:55.941 UTC
RP/0/RP0/CPU0:router(admin)# show platform

Fri Jul  9 05:45:59.805 UTCNode            Type                 PLIM             State
       Config State
-------------------------------------------------------------------------------------------
0/2/SP          MSC(SP)              N/A              UNPOWERED        NPWR,NSHUT,MON
0/RP1/CPU0      RP(Active)           N/A              IOS XR RUN       PWR,NSHUT,MON
0/SM0/SP        FC-40G/S(SP)         N/A              MBI-RUNNING      PWR,NSHUT,MON
0/SM1/*         UNKNOWN              N/A              PRESENT          PWR,NSHUT,MON

RP/0/RP1/CPU0:ios(admin)#
RP/0/RP1/CPU0:Jul  9 05:46:08.411 : instdir_lr[217]:
%INSTALL-INSTMGR-4-ACTIVE_SOFTWARE_COMMITTED_INFO :
The currently active software is not committed. If the system reboots then the committed
software will be used.
Use 'install commit' to commit the active software.
RP/0/RP1/CPU0:Jul  9 05:50:40.918 : placed[283]: LR-PLANE-READY DECLARATIONSYSTEM
CONFIGURATION COMPLETED
RP/0/RP1/CPU0:Jul  9 05:50:57.293 : ifmgr[213]: %PKT_INFRA-LINK-3-UPDOWN :
Interface MgmtEth0/RP1/CPU0/0, changed state to Down
RP/0/RP1/CPU0:Jul  9 05:50:57.313 : ifmgr[213]: %PKT_INFRA-LINK-3-UPDOWN :
Interface MgmtEth0/RP1/CPU0/0, changed state to Up
RP/0/RP0/CPU0:router(admin)# show platform

Fri Jul  9 05:59:36.266 UTC
Node            Type                 PLIM             State            Config State
-------------------------------------------------------------------------------------------
0/2/SP          MSC(SP)              N/A              UNPOWERED        NPWR,NSHUT,MON
0/RP1/CPU0      RP(Active)           N/A              IOS XR RUN       PWR,NSHUT,MON
0/SM0/SP        FC-40G/S(SP)         N/A              IOS XR RUN       PWR,NSHUT,MON
0/SM1/*         UNKNOWN              N/A              PRESENT          PWR,NSHUT,MON

RP/0/RP0/CPU0:router(admin)# install commit

Fri Jul  9 05:59:41.851 UTC
Install operation 8 '(admin) install commit' started by user 'lab' via CLI at
05:59:43 UTC Fri Jul 09 2010./
```

```
20% complete: The operation can no longer be aborted (ctrl-c for options)-
20% complete: The operation can no longer be aborted (ctrl-c for options)\
100% complete:
The operation can no longer be aborted (ctrl-c for options)
RP/0/RP1/CPU0:Jul  9 05:59:46.402 : instdir[216]:
%INSTALL-INSTMGR-4-ACTIVE_SOFTWARE_COMMITTED_INFO :
The currently active software is now the same as the committed software.
Install operation 8 completed successfully at 05:59:46 UTC Fri Jul 09 2010.
RP/0/RP0/CPU0:router(admin)# install deactivate disk0:
hfr-upgrade-p-4.0.0

Fri Jul  9 05:59:58.082 UTC
Install operation 9 '(admin) install deactivate disk0:hfr-upgrade-p-4.0.0'started
   by user 'lab' via CLI at 05:59:59 UTC
Fri Jul 09 2010.
1% complete: The operation can still be aborted (ctrl-c for options)-
1% complete: The operation can still be aborted (ctrl-c for options)
Info:     Install Method: Parallel Process Restart\
1% complete: The operation can still be aborted (ctrl-c for options)
The install operation will continue asynchronously.
RP/0/RP0/CPU0:router(admin)#
Info:     The changes made to software configurations will not be persistent
Info:     across system reloads. Use the command '(admin) install commit' to
Info:     make changes persistent.
Info:     Please verify that the system is consistent following the software
Info:     change using the following commands:
Info:         show system verify
Info:         install verify packages
RP/0/RP1/CPU0:Jul  9 06:01:45.662 : instdir[216]:
%INSTALL-INSTMGR-4-ACTIVE_SOFTWARE_COMMITTED_INFO :
The currently active software is not committed. If the system reboots then the committed
software will be used.
Use 'install commit' to commit the active software.
Install operation 9 completed successfully at 06:01:45 UTC Fri Jul 09 2010.
RP/0/RP0/CPU0:router(admin)# install commit

Fri Jul  9 06:01:53.583 UTC
Install operation 10 '(admin) install commit' started by user 'lab' via CLI at06:01:54 UTC
 Fri Jul 09 2010./
20% complete: The operation can no longer be aborted (ctrl-c for options)-
20% complete: The operation can no longer be aborted (ctrl-c for options)\
100% complete: The operation can no longer be aborted (ctrl-c for options)
RP/0/RP1/CPU0:Jul  9 06:01:57.807 : instdir[216]:
%INSTALL-INSTMGR-4-ACTIVE_SOFTWARE_COMMITTED_INFO :
The currently active software is now the same as the committed software.
Install operation 10 completed successfully at 06:01:57 UTC Fri Jul 09 2010.
RP/0/RP0/CPU0:router(admin)#
RP/0/RP0/CPU0:router(admin)#
RP/0/RP0/CPU0:router(admin)# install remove disk0:
hfr-upgrade-p-4.0.0

Fri Jul  9 06:04:57.676 UTC
Install operation 11 '(admin) install remove disk0:hfr-upgrade-p-4.0.0'started
   by user 'lab' via CLI at 06:04:58 UTC
Fri Jul 09 2010./
1% complete: The operation can no longer be aborted (ctrl-c for options)
Info:     This operation will remove the following packages:
Info:         disk0:hfr-fpd-4.0.0
Info:         disk0:hfr-doc-4.0.0
Info:         disk0:hfr-k9sec-4.0.0
Info:         disk0:hfr-sbc-4.0.0
Info:         disk0:hfr-diags-4.0.0
Info:         disk0:hfr-mgbl-4.0.0
Info:         disk0:hfr-mcast-4.0.0
```

```
Info:         disk0:hfr-mpls-4.0.0
Info:         disk0:hfr-rout-4.0.0
Info:         disk0:hfr-fwdg-4.0.0
Info:         disk0:hfr-lc-4.0.0
Info:         disk0:hfr-admin-4.0.0
Info:         disk0:hfr-upgrade-p-4.0.0-
1% complete: The operation can no longer be aborted (ctrl-c for options)
Info:     After this install remove the following install rollback point will
Info:     no longer be reachable, as the required packages will not be present:
Info:         7\
1% complete: The operation can no longer be aborted (ctrl-c for options)
Proceed with removing these packages? [confirm]|
1% complete: The operation can no longer be aborted (ctrl-c for options)
The install operation will continue asynchronously.
RP/0/RP0/CPU0:router(admin)#SP/0/SM0/SP:Jul
    9 06:05:03.902 : envmon[117]: %PLATFORM-ENVMON-4-ALARM : MINOR_HI alarm
cleared by host__temp__Inlet0
Install operation 11 completed successfully at 06:05:33 UTC
Fri Jul 09 2010.
RP/0/RP0/CPU0:router(admin)#
RP/0/RP0/CPU0:router(admin)# show install act
Fri Jul  9 06:08:11.372 UTC
Secure Domain Router: Owner  Node 0/RP1/CPU0 [HRP] [SDR: Owner]
Boot Device: disk0:    Boot Image: /disk0/hfr-os-mbi-4.0.0/mbihfr-rp.vm
Active Packages:       disk0:hfr-mpls-p-4.0.0      disk0:hfr-mini-p-4.0.0
Admin Resources:  Node 0/SM0/SP [Fabric-SP] [Admin Resource]
Boot Device: bootflash:    Boot Image: /disk0/hfr-os-mbi-4.0.0/sp/mbihfr-sp.vm
Active Packages:       disk0:hfr-mini-p-4.0.0
RP/0/RP0/CPU0:router(admin)#
```

Related Topics

Activation and Deactivation Prerequisites, on page 44
Adding and Activating Packages, on page 55
Deactivating and Removing Cisco IOS XR Software Packages, on page 72

Deactivating and Removing Cisco IOS XR Software Packages

When a package is deactivated, it is no longer active on the SDR, but the package files remain on the boot disk. The package files can be reactivated later, or they can be removed from the disk.

A package is deactivated using the following methods:

- When a newer version of a package is activated, the earlier version of the package is automatically deactivated. See *Related Topics* for more information.

Note Activating a software maintenance upgrade (SMU) does not cause any earlier SMUs or the package to which the SMU applies to be automatically deactivated.

- When an earlier version of a package is activated, the newer version is deactivated automatically. See *Related Topics* for more information.
- A specific package is deactivated using the **install deactivate** command. This command turns off the package features for a card or card type.

Before you begin

The following are the restrictions when deactivating and removing Cisco IOS XR Software packages:

- A package cannot be deleted if it is part of the running or committed software of the SDR.
- A package cannot be deactivated if that package is required by another active package. When a deactivation is attempted, the system runs an automatic check to ensure that the package is not required by other active packages. The deactivation is permitted only after all compatibility checks have been passed.
- Router reloads: If the deactivation requires a router reload, a confirmation prompt appears. Use the **install deactivate** command with the **prompt-level none** keywords to automatically ignore any reload confirmation prompts and proceed with the package deactivation. The router reloads if required.
- Node reloads: If a software operation requires a node reload, the configuration register for that node should be set to autoboot. If the config-register for the node is not set to autoboot, then the system automatically changes the setting and the node reloads. A message describing the change is displayed.
- FPD versions must be compatible with the Cisco IOS XR software that is running on the router; if an incompatibility exists between an FPD version and the Cisco IOS XR software, the device with the field-programmable gate array (FPGA) may not operate properly until the incompatibility is resolved. For information on FPDs, including instructions to upgrade FPD images, see the *Upgrading FPD Cisco IOS XR Software* module of *Interface and Hardware Component Configuration Guide for Cisco CRS Routers*.

SUMMARY STEPS

1. Connect to the console port and log in.
2. **admin**
3. **install deactivate** { **id** *add-id* | *device* **:** *package* } [**location** *node-id*] [**test**] [**pause sw-change**]
4. (Optional) **show install inactive summary**
5. (Optional) **install verify packages**
6. **exit**
7. (Optional) **show system verify start**
8. (Optional) **show system verify** [**detail** | **report**]
9. **admin**
10. (Optional) **install commit**
11. (Optional) **install remove** { **id** *add-id* | *device* **:** *package* | **inactive** } [**test**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	Connect to the console port and log in.	Establishes a CLI management session with the SDR. Connect to the console port for the active DSC. For more information on console connections, see *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*.
Step 2	Required: **admin** **Example:**	Enters administration EXEC mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router# admin`	
Step 3	**install deactivate** { **id** *add-id* | *device* **:** *package* } [**location** *node-id*] [**test**] [**pause sw-change**] **Example:** `RP/0/RP0/CPU0:router(admin)# install deactivate disk0:hfr-diags-3.8.30`	Deactivates a package on all SDRs . • To deactivate all packages that were added in one or more specific install add operations, or specify packages by name, use the **id** *add-id* keyword and argument. The operation ID of an install add operation is indicated in the syslog displayed during the operation and in the output of the **show install log** command. • Use the **location** *node-id* keyword and argument to deactivate the package for a specific node, if supported. • Use the **pause sw-change** keywords to pause the operation after preparatory checks and before the configuration is locked for the actual deactivation. This enables you to hold the operation while you perform configuration changes, and proceed with the deactivation whenever you choose. This is useful, for example, if your workflow involves configuring a router out of the network during software changes and you want to minimize the time that the router is out of the network. Follow the onscreen instructions to control the pausing and completion of the operation. **Note** Press **?** after a partial package name to display all possible matches available for deactivation. If there is only one match, press [TAB] to fill in the rest of the package name. When a package is deactivated for an SDR from administration EXEC mode, a notification message appears on the console for that SDR, with information on the impact of the deactivation.
Step 4	(Optional) **show install inactive summary** **Example:** `RP/0/RP0/CPU0:router(admin)# show install inactive summary`	Displays the inactive packages on the router.
Step 5	(Optional) **install verify packages** **Example:**	Verifies the consistency of an installed software set with the package file from which it originated. This command can be used as a debugging tool to verify the validity of

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(admin)# install verify packages`	the files that constitute the packages, to determine if there are any corrupted files. This command also checks for corruptions of installation state files and MBI image files. This command is particularly useful when issued after the activation of a package or upgrading the Cisco IOS XR software to a major release. **Note** The **install verify packages** command can take up to two minutes per package to process.
Step 6	Required: **exit** **Example:** `RP/0/RP0/CPU0:router(admin)# exit`	Exits administration EXEC mode and returns to EXEC mode.
Step 7	(Optional) **show system verify start** **Example:** `RP/0/RP0/CPU0:router# show system verify start`	Starts the system status check.
Step 8	(Optional) **show system verify** [**detail** | **report**] **Example:** `RP/0/RP0/CPU0:router# show system verify`	Displays system status information. A variety of information is displayed including the memory and CPU usage, process status, protocol status, and other status information. Use this information to verify that the system is stable. • **detail**—Displays additional information at the card and processor level, including actual numbers. • **report**—Displays the same information as the default **show system verify** command **Note** Although most of the output should display the status "OK," some processes may show other output, such as "Warning." This does not specifically indicate a problem. Contact your Cisco technical support representative for more information on the output of this command.
Step 9	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 10	(Optional) **install commit** **Example:** `RP/0/RP0/CPU0:router(admin)# install commit`	Commits the current set of packages so that these packages are used if the router is restarted. Packages can be removed only if the deactivation operation is committed. **Note** This command is entered in administration EXEC mode.

	Command or Action	Purpose
Step 11	(Optional) **install remove** { **id** *add-id* | *device* **:** *package* | **inactive** } [**test**] **Example:** `RP/0/RP0/CPU0:router(admin)# install remove disk0:hfr-diags-3.8.30`	Removes the inactive package. • Only inactive packages can be removed. • Packages can be removed only if they are deactivated from all cards in all SDRs . • The package deactivation must be committed. • To remove a specific inactive package from a storage device, use the **install remove** command with the device**:** *package* arguments. • To remove all packages that were added in one or more specific **install add** operations, use the **id** *add-id* keyword and argument. The operation ID of an **install add** operation is indicated in the syslog displayed during the operation and in the output of the **show install log** command. If you specify packages according to operation ID, all the packages that were added by the specified operation must still be on the router. • To remove all inactive packages from all nodes in the system, use the **install remove** command with the **inactive** keyword. • To remove packages from all SDRs, use the **install remove** command in administration EXEC mode.

Related Topics

Adding and Activating Packages, on page 55
Committing the Active Package Set, on page 64

Examples

In the following examples, a package is deactivated from all SDRs in a router . The changes are committed and the inactive package is removed from the router.

Deactivating the Package: Example

```
RP/0/RP0/CPU0:router(admin)# install deactivate disk0:hfr
-diags-.7.2

Install operation 27 'install deactivate disk0:hfr-diags-3.7.2' started by
user 'lab' at 23:29:37 UTC Sat Apr 15 2009.
The install operation will continue asynchronously.
Info:     The changes made to software configuration
Info:     across system reloads. Use the command 'admin install commit' to make
Info:     changes persistent.
Info:     Please verify that the system is consistent following the software
Info:     change using the following commands:
Info:         show system verify
Info:         install verify packages
Install operation 27 completed successfully at 23:30:22 UTC Sat Apr 15 2009.
```

Committing the Active Software Set: Example

```
RP/0/RP0/CPU0:router(admin)# install commit

Install operation 29 'install commit' started by user 'lab' at 23:39:21 UTC
Sat Apr 15 20090.
Install operation 29 completed successfully at 23:39:24 UTC Sat Apr 15 2009.
```

Displaying the Inactive Packages: Example

```
RP/0/RP0/CPU0:router(admin)# show install inactive summary

Default Profile:
  SDRs:
  Owner
  Inactive Packages:
    disk0:hfr-diags-3.7.2
```

Removing the Inactive Package from the Router: Example

The following example shows how to remove an inactive package. In this example, the operation is run in test mode. The operation is confirmed and the package is removed.

```
RP/0/RP0/CPU0:router(admin)# install remove disk0:hfr-diags-3.7.2 test

Install operation 30 'install remove disk0:hfr-diags-3.7.2 test' started by
user 'lab' at 23:40:22 UTC Sat Apr 15 2009.
Warning:  No changes will occur due to 'test' option being specified. The
Warning:  following is the predicted output for this install command.
Info:     This operation will remove the following package:
Info:         disk0:hfr-diags-3.7.2
Info:     After this install remove the following install rollback points will
Info:     no longer be reachable, as the required packages will not be present:
Info:         4, 9, 10, 14, 15, 17, 18
Proceed with removing these packages? [confirm] y

The install operation will continue asynchronously.
Install operation 30 completed successfully at 23.
```

Pausing Before Configuration Lock: Example

The following example shows how to deactivate a package, pausing the operation before locking the configuration for the actual software deactivation. While the operation is paused, you can enter a configuration mode and perform configurations. When you want to complete the operation, enter the **install operation** *id* **complete** command, or the **install operation** *id* **attach synchronous** command.

```
RP/0/RP0/CPU0:router(admin)# install deactivate disk0:comp-hfr
-3.7.2.07I.CSCsr09575-1.0.0 pause sw-change

Install operation 12 '(admin) install deactivate
  disk0:comp-hfr-3.7.2.07I.CSCsr09575-1.0.0 pause sw-change'
  started by user 'admin' via CLI at 09:06:26 BST Mon Jul 07 2009.
Info: This operation will reload the following nodes in parallel:
Info: 0/0/CPU0 (RP) (SDR: Owner)
Info: 0/1/CPU0 (LC(E3-GE-4)) (SDR: Owner)
```

```
Info: 0/5/CPU0 (LC(E3-OC3-POS-4)) (SDR: Owner)
Proceed with this install operation (y/n)? [y]
The install operation will continue asynchronously.
Info: Install Method: Parallel Reload
Info: Install operation 12 is pausing before the config lock is applied for
Info:    the software change as requested by the user.
Info: No further install operations will be allowed until the operation is resumed.
Info: Please continue the operation using one of the following steps:
Info: - run the command '(admin) install operation 12 complete'.
Info: - run the command '(admin) install operation 12 attach synchronous' and then
Info:      answer the query.
```

Rolling Back to a Previous Software Set

Cisco IOS XR software allows you to roll back one or more SDRs to a previous committed or uncommitted software set. Use the **show install rollback ?** command to view the available rollback points and use the **install rollback to** command to roll back the SDR to a previous software set. You can also use the **install rollback to committed** command to roll back to the most recent committed software set.

Note Rollback operations can be performed for all SDRs by running the command in administration EXEC or for a single SDR by running the command in either administration EXEC or EXEC mode.

Displaying Rollback Points

A rollback point is created every time a software package is activated, deactivated, or committed. Use the **show install rollback ?** command to display the eligible rollback points.

```
RP/0/RP0/CPU0:router# admin
RP/0/RP0/CPU0:router(admin)# show install rollback ?

    0  ID of the rollback point to show package information for
    2  ID of the rollback point to show package information for
```

In this example, the rollback points are 0 and 2. The rollback point with the highest number is the current software point. For example, if the last installation operation was operation 3 (activating the MPLS package) then the highest rollback point is 3, which is the same as the current software (MPLS package activated).

To easily identify specific rollback points, you can assign a label or description to a rollback point using the **install label** command.

Enter the command in administration EXEC mode to display rollback points for all SDRs. Enter the command in EXEC mode to display rollback points for the SDR to which you are currently logged in. You can also display rollback points for a specific SDR in administration EXEC mode by using the **sdr** *sdr-name* keyword and argument.

Displaying the Active Packages Associated with a Rollback Point

To display the active packages associated with a rollback point, use the **show install rollback** command with the *point-id* argument. This command displays the packages that are active if you roll back one or more

SDRs to that installation point. For example, the **show install rollback 2** command displays the packages that are active if you roll back to rollback point 2.

```
RP/0/RP0/CPU0:router(admin)# show install rollback 2

  Secure Domain Router: Owner

    Node 0/1/SP [SP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.8.84/sp/mbihfr-sp.vm
      Rollback Packages:
        disk0:hfr-diags-3.8.84
        disk0:comp-hfr-mini-3.8.84

    Node 0/1/CPU0 [LC] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.8.84/lc/mbihfr-lc.vm
      Rollback Packages:
        disk0:hfr-diags-3.8.84
        disk0:comp-hfr-mini-3.8.84

    Node 0/RP0/CPU0 [RP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.8.84/mbihfr-rp.vm
      Rollback Packages:
        disk0:hfr-diags-3.8.84
        disk0:hfr-mgbl-3.8.84
        disk0:hfr-k9sec-3.8.84
        disk0:comp-hfr-mini-3.8.84

    Node 0/RP1/CPU0 [RP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.8.84/mbihfr-rp.vm
      Rollback Packages:
        disk0:hfr-diags-3.8.84
        disk0:hfr-mgbl-3.8.84
        disk0:hfr-k9sec-3.8.84
        disk0:comp-hfr-mini-3.8.84

    Node 0/SM0/SP [SP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.8.84/sp/mbihfr-sp.vm
      Rollback Packages:
        disk0:hfr-diags-3.8.84
        disk0:comp-hfr-mini-3.8.84
  Node 0/SM1/SP [SP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.8.84/sp/mbihfr-sp.vm
      Rollback Packages:
        disk0:hfr-diags-3.8.84
        disk0:comp-hfr-mini-3.8.84

    Node 0/SM2/SP [SP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.8.84/sp/mbihfr-sp.vm
      Rollback Packages:
        disk0:hfr-diags-3.8.84
        disk0:comp-hfr-mini-3.8.84

    Node 0/SM3/SP [SP] [SDR: Owner]
      Boot Image: /disk0/hfr-os-mbi-3.8.84/sp/mbihfr-sp.vm
      Rollback Packages:
        disk0:hfr-diags-3.8.84
        disk0:comp-hfr-mini-3.8.84
```

Enter the command in administration EXEC mode to view the rollback points for all SDRs. Enter the command in EXEC mode to display the rollback points for the SDR to which you are currently logged in. You can also

display rollback points for a specific SDR in administration EXEC mode by using the **sdr** *sdr-name* keyword and argument.

Note For more information on the command options, see the *Software Package Management Commands on Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*.

Rolling Back to a Specific Rollback Point

You can roll back to a specific rollback point, including a noncommitted software set:

- If you roll back to the most recent noncommitted rollback point (with the highest number), you do not need to reload the router.
- You can repeat the rollback process one rollback point at a time without reloading if you always choose the most recent rollback point.
- If you choose a rollback point that is older than the most recent point, the impacted nodes reload, interrupting data traffic on those nodes. Before the reload occurs, you are prompted to confirm the install rollback operation.

In the following example, the system is rolled back to noncommitted rollback point 8:

```
RP/0/RP0/CPU0:router(admin)# install rollback to 8

  Install operation 10 'install rollback to 8' started by user 'cisco' at 07:49:26
  UTC Mon Nov 14 2009.
  The install operation will continue asynchronously.
  Info:     The changes made to software configurations will not be persistent
  Info:     across system reloads. Use the command 'admin install commit' to make
  Info:     changes persistent.
  Info:     Please verify that the system is consistent following the software
  Info:     change using the following commands:
  Info:         show system verify
  Info:         install verify packages

  The currently active software is the same as the committed software.

  Install operation 10 completed successfully at 07:51:24 UTC Mon Nov 14 2009.
```

Rolling Back to the Last Committed Package Set

Use the **install rollback to committed** command to roll back to the last committed package set.

In the following example, all SDRs in the system are rolled back to the last committed package set :

```
RP/0/RP0/CPU0:router(admin)# install rollback to committed

  Install operation 27 'install rollback to committed' started by user 'lab' at
  16:41:38 UTC Sat Nov 19 2009.
  Info:     The rollback to committed software will require a reload of impacted
  Info:     nodes because it is over multiple activation & deactivation
  Info:     operations.
  Info:     This operation will reload the following node:
  Info:         0/RP1/CPU0 (RP) (SDR: Owner)
  Info:     This operation will reload all RPs in the Owner SDR, and thereby
```

```
Info:     indirectly cause every node in the router to reload.

Proceed with this install operation? [confirm]

Updating Commit Database.  Please wait...[OK]
Info:     The changes made to software configurations will not be persistent
Info:     across system reloads. Use the command 'admin install commit' to make
Info:     changes persistent.
Info:     Please verify that the system is consistent following the software
Info:     change using the following commands:
Info:         show system verify
Info:         install verify packages
Install operation 27 completed successfully at 16:42:23 UTC Sat Nov 19 2009.
```

Additional References

The following sections provide references related to software package management on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Cisco IOS XR install commands	*Software Package Management Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*
Cisco IOS XR getting started material	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*
Cisco IOS XR master command index	*Cisco IOS XR Commands Master List for the Cisco CRS Router*
Information about user groups and task IDs	*Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*
ROM Monitor	*ROM Monitor Configuration Guide for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

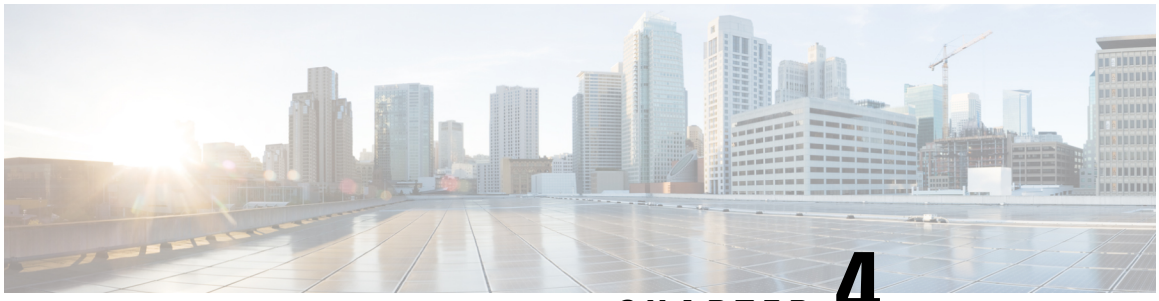
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 4

Configuring Disk Backup and Disk Mirroring

This module describes the process to configure disk mirroring and create a backup disk of the packages and configurations in Cisco IOS XR software.

Note The disk backup feature is also known as *Golden Disk*.

For complete descriptions of the commands listed in this module, see Related Documents, on page 101. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 9: Feature History for Disk Backups and Disk Mirroring for Cisco IOS XR Software

Release	Modification
Release 3.4.0	Backup disk creation was introduced.
Release 3.6.0	Disk mirroring was introduced.

This module contains the following topics:

Disk Backup Prerequisites

Before performing a system backup, the following conditions must be met:

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- Local storage device specified for the backup must be installed. The supported storage devices are disk0: and disk1: (if installed).
- Disk mirroring must not be enabled.

Disk Mirroring Prerequisites

Before enabling disk mirroring, the following conditions must be met:

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- The secondary storage device specified for the mirroring must be installed in the same node as the primary boot device. The supported storage devices are disk0: and disk1:.
- The secondary storage device must be the same size or larger than the designated primary storage device.
- The secondary storage device must be partitioned.
- After disk mirroring is enabled, you cannot configure the backup disk feature.

Note The primary partition on the secondary storage device must be large enough to contain all data on the primary boot device. This can be an issue if the primary boot device has not yet been partitioned. For example, in the situation where both the primary boot device and the secondary storage device are 1 GB in size, the primary boot device contains 950 MB of data, and the secondary storage device is already partitioned to 800 MB in the primary partition and 200 MB in the secondary partition. In such a case, the 950 MB of data from the primary boot device does not fit on the secondary storage device because of the partition. Such a configuration is rejected and an error is displayed. You need to replace the secondary storage device with a higher capacity device. For information about disk partition sizes, see *Related Topics*.

Note Although compactflash: can be used as the secondary device on a Performance Route Processor (PRP–2), there is an issue with the ROM Monitor not being able to boot the minimum boot image (MBI) from the secondary device if the device is not disk0: or disk1:. In such a situation, you would need to go into ROMMON mode and boot the PRP-2 manually using the MBI on the compactflash:.

Related Topics

Information About Disk Mirroring, on page 85

Information About the Backup Disk

A system backup disk is created when you back up the system files to a local storage device for the first time. This process formats the selected device, and copies the software packages and system configurations to that device. If the backup operation is performed from EXEC mode, then the files from that specific secure domain

router (SDR) are backed up. If the backup operation is performed from administration EXEC mode, then the files from the administration plane and from all SDRs are backed up .

Before you create a backup disk of the Cisco IOS XR software packages and configurations, you need to determine which device is being used as the primary boot device. The boot device is displayed using the following commands:

- **show version**
- **show install active**
- **show install committed**

See *System Management Command Reference for isco CRS Routers* for information on using the commands to determine the boot device.

Information About Disk Mirroring

The route processor (RP) card has a primary storage device that is used to store installation packages and configuration files. This primary storage device is referred to as the *primary boot device* and is essential for booting the RP and its normal operation.

Disk mirroring replicates the critical data on the primary boot device onto another storage device on the same RP, henceforth referred to as the secondary device. If the primary boot device fails, applications continue to be serviced transparently by the secondary device, thereby avoiding a switchover to the standby RP. The failed primary storage device can be replaced or repaired without disruption of service.

Disk mirroring should only mirror critical data on the primary boot device onto a secondary storage device and not any noncritical data such as logging data. To separate critical data from noncritical data, the disk devices need to be partitioned. Disk0: is partitioned to disk0: and disk0a:; disk1: is partitioned to disk1: and disk1a:. Disk0: and disk1: are used for critical data, whereas disk0a: and disk1a: are used for logging data and other noncritical data. Before you can configure disk mirroring on the RP, you must have partitioned the secondary storage device. The sizes of disk partitions are related to the total disk size, and are provided in Table 10: Size of Disk Partitions in Relation to Size of Disk, on page 85.

Table 10: Size of Disk Partitions in Relation to Size of Disk

Size of Disk	Primary Partition Percentage	Secondary Partition Percentage
less than 900 MB	Partitioning not supported	Partitioning not supported
900 MB to 1.5 GB	80%	20%
1.5 GB to 3 GB	60%	40%
more than 3 GB	50%	50%

How to Create a Backup Disk

Creating a New or Revised Backup Disk

Complete the following instructions to create a new backup disk, or to back up the most recent software and configuration files onto an existing backup disk.

Note If disk mirroring is enabled, you will not be able to configure a backup disk.

SUMMARY STEPS

1. **admin**
2. (Optional) **showsystem backup** [*target-dev*] [**details** | **diff**] [**verify**] [**location** {**all** | *node-id*}]
3. (Optional) **system boot-sequence** *primary-device* [*secondary-device*] [**location** {**all** | *node-id*}]
4. **system backup** [*target-dev*] [**format**] [**location** {**all** | *node-id*}] [**synchronous** | **asynchronous**]
5. (Optional) **show system backup** [*target-dev*] [**details** | **diff**] [**verify**] [**location** {**all** | *node-id*}]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	(Optional) **showsystem backup** [*target-dev*] [**details**	**diff**] [**verify**] [**location** {**all**	*node-id*}] **Example:** `RP/0/RP0/CPU0:router# show system backup disk1:`	Displays information for the backup performed on the active RP where you are logged in, including the date, time, and status of the last backup. Use this information to determine if a backup is required. The following options display specific information: • *target-dev*—Displays backup information for a specified RP node. • **details**—Lists information about the software packages and configuration files stored on the backup device. • **diff**—Displays the differences between the software and configurations on the backup device and the software and configurations on the currently active boot disk. • **location** *node-id*—Displays information for a backup on a specific node. Use the **location all** keywords to display information for backups on all nodes in the system. If no backup exists, an error message is displayed.

	Command or Action	Purpose
Step 3	(Optional) **system boot-sequence** *primary-device* [*secondary-device*] [**location** {**all** | *node-id*}] **Example:** `RP/0/RP0/CPU0:router(admin)# system boot-sequence disk0: disk1:`	Defines the order of local storage devices used to boot a router. Enter a value for the secondary device field to define the default location for system backups. If this field is left blank, the backup device can be defined with the **system backup** command, as shown in the next step. • The boot devices specified in this command must be installed in the card, or the command will fail. • *primary-device*—Defines the default device where software packages are installed and run. This device is also the default location for router configurations. We recommend disk0: as the primary boot device in the boot sequence. • *secondary-device*—Defines the device used by the **system backup** command to back up system software and configurations. The value of the *secondary-device* argument must be different from the value of the *primary-device* argument. We recommend disk1: as the secondary boot device in the boot sequence. This field is optional. • **location** *node-id*—Defines the boot sequence for a specific RP. • **location all**—Defines the boot sequence for all RPs in the SDR. Use this command in administration EXEC mode to define the boot sequence for all RPs in all SDRs. The *node-id* argument is expressed in *rack*/*slot*/*module* notation.
Step 4	**system backup** [*target-dev*] [**format**] [**location** {**all** | *node-id*}] [**synchronous** | **asynchronous**] **Example:** `RP/0/RP0/CPU0:router(admin)# system backup disk0: disk1:`	Backs up the system software and configurations to a backup disk. Use the **system backup** command without keywords or arguments to back up the system software and configurations on the target device for the designated secure domain router shelf controller (DSDRSC) where you are logged in. • By default, the backup disk is the secondary device defined with the **system boot-sequence** command in Step 3, on page 87. • To define a backup device for the current backup operation only, use the **system backup** command with the *target-dev* argument. • If a target device is not specified with either the **system backup** *target-dev* command or the **system boot-sequence** command, then the backup operation is not allowed.

	Command or Action	Purpose
		Note The *target-dev* argument can be any local storage device except the current boot device, and must be large enough to store the current software set and configuration. • **location** *node-id*—Specifies an alternate node for the system backup, such as the standby DSDRSC . • **location all**—Performs the backup on all RPs installed in a specific SDR, when used in EXEC mode; performs the backup on all RPs in all SDRs installed in the system, when used in administration EXEC mode. • Default is **synchronous**.
Step 5	(Optional) **show system backup** [*target-dev*] [**details** | **diff**] [**verify**] [**location** {**all** | *node-id*}] **Example:** `RP/0/RP0/CPU0:router# show system backup disk1:`	Displays information for the backup performed on the active RP where you are logged in, including the date, time, and status of the last backup. Use this information to determine if a backup is required. The following options display specific information: • *target-dev*—Displays backup information for a specified RP node. • **details**—Lists information about the software packages and configuration files stored on the backup device. • **diff**—Displays the differences between the software and configurations on the backup device and the software and configurations on the currently active boot disk. • **location** *node-id*—Displays information for a backup on a specific node. Use the **location all** keywords to display information for backups on all nodes in the system. If no backup exists, an error message is displayed.

Using the Backup Disk to Boot a Router

Complete the following instructions to use the backup disk to boot a router.

SUMMARY STEPS

1. Enter ROM Monitor mode.
2. **dir disk1:**
3. Locate the hfr-os-mbi-X.Y.Z directory.
4. **dir disk1: hfr -os-mbi-** *X* **.** *Y* **.** *Z*
5. Locate the mbihfr- *xx* .vm file.
6. **unset BOOT**
7. **sync**
8. **confreg 0x102**

9. **boot disk1: hfr -os-mbi-***X***.***Y***.***Z***/mbi hfr hfr**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Enter ROM Monitor mode.	See *ROM Monitor Configuration Guide for Cisco CRS Routers* for information on entering ROM Monitor mode.
Step 2	**dir disk1:** **Example:** `rommon1> dir disk1:`	Displays the files on the disk1: storage device.
Step 3	Locate the hfr-os-mbi-X.Y.Z directory.	Identifies the directory on the disk1: storage device.
Step 4	**dir disk1: hfr -os-mbi-***X***.***Y***.***Z* **Example:** `rommon2> dir disk1:hfr -os-mbi-3.8.0`	Displays the files in the hfr-os-mbi-X.Y.Z directory.
Step 5	Locate the mbihfr-*xx*.vm file.	Identifies the file in the hfr-os-mbi-X.Y.Z directory. **Note** On the RP, *xx* = **rp** ; on the DRP, *xx* = **drp** .
Step 6	**unset BOOT** **Example:** `rommon3> unset BOOT`	Clears the setting for the BOOT variable.
Step 7	**sync** **Example:** `rommon4> sync`	Saves the new ROM Monitor variable settings.
Step 8	**confreg 0x102** **Example:** `rommon5> confreg 0x102`	Sets the configuration register to 0x102.
Step 9	**boot disk1: hfr -os-mbi-***X***.***Y***.***Z***/mbi hfr hfr** **Example:** `rommon6> boot disk1:hfr-os-mbi-3.8.0/mbihfr-rp.vm`	Retrieves the file and installs it on the boot device. **Note** On the RP, *xx* = **rp** ; on the DRP, *xx* = **drp** .

How to Enable Disk Mirroring

The tasks in this section describe how to enable and manage disk mirroring.

Enabling Disk Mirroring

Complete the following instructions to enable disk mirroring. After disk mirroring is configured, if there is a fault on the primary boot drive or it cannot be accessed for any reason, control is automatically transferred to the secondary storage device.

Note Disk mirroring overrides any existing disk backup configuration (Golden Disk), and subsequent disk backups fail.

SUMMARY STEPS

1. **format** *secondary-device* **partition** [**location** *node-id*]
2. Remove any noncritical data from the primary boot device.
3. **configure**
4. **mirror location** *node-id Primary-device Secondary-device*
5. **commit**
6. **show mirror** [**location** *node-id*]
7. **mirror verify location** *node-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**format** *secondary-device* **partition** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# format disk1: partition`	Partitions the secondary storage device into two partitions. • If the device is already partitioned, you do not need to perform this step. • On multishelf systems, specify the node of the primary boot device with the **location** keyword. **Note** When you partition a FAT16 disk, the file system changes to FAT32. If you might need to downgrade to Cisco IOS XR Software Release 3.7.0 or earlier, consider carefully before partitioning the disk. Software downgrades to Cisco IOS XR Software Release 3.7.0 or earlier are not supported for FAT32 disks. The procedure for converting a FAT32 disk to FAT16 is complex and requires the assistance of Cisco technical support.
Step 2	Remove any noncritical data from the primary boot device.	The primary boot device should contain installation packages and configuration files only. Log files can be copied to the “a” partition of the secondary device, for example disk1a: .
Step 3	**configure**	

	Command or Action	Purpose
Step 4	**mirror location** *node-id Primary-device Secondary-device* **Example:** `RP/0/RP0/CPU0:router(config)# mirror location 0/rp 0/cpu0 disk0:disk1:`	Enables disk mirroring of the *primary-device* to the *secondary-device* . On multishelf systems, specify the node of the primary boot device with the **location** keyword. If the primary boot device is not partitioned, the following occurs: • The contents of the primary device are replicated to the secondary device • Control of the mirroring server switches to the secondary storage device. • The primary device is partitioned. • Data is replicated back to the primary boot device.
Step 5	**commit**	
Step 6	**show mirror** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show mirror location 0/rp 0/cpu0`	Displays disk mirroring information for an RP node. It also provides the status of the synchronization between the primary and secondary devices. Use the **location** *node-id* keyword and argument to view the mirror status on other line card and fabric card chassis. The default display is the mirror status of the card to which the console is connected.
Step 7	**mirror verify location** *node-id* **Example:** `RP/0/RP0/CPU0:router# mirror verify location 0/rp 0/cpu0`	Verifies disk synchronization for disk mirroring on an RP node.

Replacing the Secondary Mirroring Device

Follow this procedure if you need to replace the secondary boot device used in the disk mirroring process.

SUMMARY STEPS

1. **show mirror** [**location** *node-id*]
2. **mirror pause** [**location** *node-id*]
3. **show mirror** [**location** *node-id*]
4. **unmount** *secondary-device* [**location** *node-id*]
5. Remove the device and insert a new device.
6. **format** *secondary-device* **partition** [**location** *node-id*]
7. **show media** [**location** *node-id*]
8. **mirror resume** [**location** *node-id*]
9. **show mirror** [**location** *node-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show mirror** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show mirror`	Verifies that mirroring is active. In the output, the *Current Mirroring State* should be redundant. Use the **location** *node-id* keyword and argument to view the mirror status on other line card and fabric card chassis. The default display is the mirror status of the card to which the console is connected.
Step 2	**mirror pause** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# mirror pause`	Temporarily pauses disk mirroring.
Step 3	**show mirror** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show mirror`	Verifies that mirroring has paused. In the output, the *Current Mirroring State* should be paused.
Step 4	**unmount** *secondary-device* [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# unmount disk1:`	Unmounts the secondary device.
Step 5	Remove the device and insert a new device.	
Step 6	**format** *secondary-device* **partition** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# format disk1: partition`	Formats the device.
Step 7	**show media** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show media`	Verifies that the device is formatted. The output should display the device that you formatted.
Step 8	**mirror resume** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# mirror resume`	Resumes mirroring.
Step 9	**show mirror** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show mirror`	Verifies that mirroring has restarted. In the output, the *Current Mirroring State* should be Syncing. It can take 15 to 30 minutes for the mirroring process to complete. The exact time depends on the number of packages or files on the boot device. When the mirroring is complete, the *Current Mirroring State* should be Redundant.

Replacing the Primary Mirroring Device

In the event that your primary boot disk is defective and you need to replace it while disk mirroring is enabled, perform this task.

SUMMARY STEPS

1. **show mirror** [**location** *node-id*]
2. **configure**
3. **mirror location** *node-id Primary-device Secondary-device*
4. **commit**
5. **show mirror** [**location** *node-id*]
6. **mirror pause** [**location** *node-id*]
7. **show mirror**
8. **unmount** *secondary-device* [**location** *node-id*]
9. Remove the device and insert a new device.
10. **show media** [**location** *node-id*]
11. (Optional) **format** *secondary-device* **partition** [**location** *node-id*]
12. **mirror resume** [**location** *node-id*]
13. **show mirror** [**location** *node-id*]
14. **configure**
15. **mirror location** *node-id Primary-device Secondary-device*
16. **show mirror** [**location** *node-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show mirror** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show mirror`	Verifies that mirroring is in the redundant state. In the output, the *Current Mirroring State* should be redundant. If mirroring is not in the redundant state, you cannot proceed with the procedure. You must wait until mirroring is in the redundant state. Use the **location** *node-id* keyword and argument to view the mirror status on other line card and fabric card chassis. The default display is the mirror status of the card to which the console is connected.
Step 2	**configure**	
Step 3	**mirror location** *node-id Primary-device Secondary-device* **Example:** `RP/0/RP0/CPU0:router(config)# mirror location 0/RP0 /CPU0 disk1:disk0:`	Swaps the device roles such that the primary mirroring device now becomes the secondary device and the secondary mirroring device becomes the primary device.

	Command or Action	Purpose
Step 4	**commit**	
Step 5	**show mirror** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show mirror`	Verifies that the primary device is now the secondary device and vice versa. In the output, if disk0: was the primary disk that you want to replace, it should now be listed as the secondary device.
Step 6	**mirror pause** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# mirror pause`	Temporarily pauses disk mirroring.
Step 7	**show mirror** **Example:** `RP/0/RP0/CPU0:router# show mirror`	Verifies that mirroring has paused. In the output, the *Current Mirroring State* should be paused.
Step 8	**unmount** *secondary-device* [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# unmount disk1:`	Unmounts the secondary device which is the device that you want to replace. Initially, this was the primary device.
Step 9	Remove the device and insert a new device.	
Step 10	**show media** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show media`	Verifies that the new disk is partitioned. You should see that the new device is mounted. If the new device is not partitioned, format the device as indicated in the next step.
Step 11	(Optional) **format** *secondary-device* **partition** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# format disk1: partition`	Formats the device. You only need to perform this step if the new device is not partitioned.
Step 12	**mirror resume** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# mirror resume`	Resumes mirroring.
Step 13	**show mirror** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show mirror`	Verifies that mirroring has restarted. In the output, the *Current Mirroring State* should be Syncing. It can take 15 to 30 minutes for the mirroring process to complete. The exact time depends on the number of packages or files on the boot device. When the mirroring is complete, the *Current Mirroring State* should be Redundant.

	Command or Action	Purpose
Step 14	**configure**	
Step 15	**mirror location** *node-id Primary-device Secondary-device* **Example:** `RP/0/RP0/CPU0:router(config)# mirror location 0/RP0 /CPU0 disk0:disk1:`	Swaps the device roles back so that the newly inserted device becomes the primary device.
Step 16	**show mirror** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show mirror`	Verifies that the new device is now the primary device.

Configuration Examples for Creating a Backup Disk

Defining the Boot Disk Sequence: Example

The following example shows how to define the primary and secondary boot device for the active RP (or DSC). The secondary device is also the default location for system backups. In this example, the default location for software and configurations is disk0:. The location for backups of software and configurations is disk1:.

```
admin
  system boot-sequence disk0: disk1:
  Info:  node0_0_CPU0: command succeeded.
```

Creating a Backup Disk for All SDRs : Example

The following example shows how to back up the software and configuration files:

- The command is run in administration EXEC mode, which backs up both the administration and SDR configurations.
- The target device is defined by the value entered with the **system boot-sequence** command, as shown in Defining the Boot Disk Sequence: Example, on page 95.
- Because this is the first backup on the device, the disk is formatted.

```
admin
  system backup location all

  Info: node0_0_CPU0: formatting target device
  Info: node0_1_CPU0: formatting target device
  Info: node0_3_CPU0: formatting target device
  Info: node0_0_CPU0: copying admin configuration
```

```
  Info: node0_1_CPU0: copying admin configuration
  Info: node0_3_CPU0: copying SDR configuration
  Info: node0_0_CPU0: copying SDR configuration
  Info: node0_3_CPU0: copying installed software
  Info: node0_1_CPU0: copying SDR configuration
  Info: node0_0_CPU0: copying installed software
  Info: node0_1_CPU0: copying installed software
  Info: node0_0_CPU0: backup complete.
  Info: node0_3_CPU0: backup complete.
  Info: node0_1_CPU0: backup complete.
  Info: node0_0_CPU0: verifying admin configuration
  Info: node0_1_CPU0: verifying admin configuration
  Info: node0_3_CPU0: verifying SDR configuration
  Info: node0_0_CPU0: verifying SDR configuration
  Info: node0_1_CPU0: verifying SDR configuration
  Info: node0_3_CPU0: verifying installed software
  Info: node0_0_CPU0: verifying installed software
  Info: node0_1_CPU0: verifying installed software
  Info: node0_3_CPU0: verify complete.
  Info: node0_3_CPU0: command succeeded.
  Info: node0_1_CPU0: verify complete.
  Info: node0_1_CPU0: command succeeded.
  Info: node0_0_CPU0: verify complete.
  Info: node0_0_CPU0: command succeeded.
```

Creating a Backup Disk for a Single SDR: Example

In the following example, the backup disk is created for a non-owner SDR.

- The command is run in EXEC mode, which backs up only the current SDR files and configuration.
- The target device is defined as disk1:.
- Because this is the first backup on the device, the disk is formatted.

```
system backup disk1:

  Info:  node0_3_CPU0: formatting target device
  Info:  node0_3_CPU0: copying SDR configuration
  Info:  node0_3_CPU0: copying installed software
  Info:  node0_3_CPU0: backup complete.
  Info:  node0_3_CPU0: verifying SDR configuration
  Info:  node0_3_CPU0: verifying installed software
  Info:  node0_3_CPU0: verify complete.
  Info:  node0_3_CPU0: command succeeded.
```

Showing the Backup Information: Examples

In the following example, the **show system backup** command displays the status of the last system backup:

```
RP/0/RP0/CPU0:router# admin
RP/0/RP0/CPU0:router(admin)# show system backup

System Backup information for node0_0_CPU0 on disk1:
=====================================================
  Last Backup Successful
```

```
  Backup started at Sat Jun 24 12:22:10 2009
   ended at Sat Jun 24 12:42:11 2009
  Verify started at Sat Jun 24 12:42:12 2009
   ended at Sat Jun 24 12:48:47 2009
  BOOT_DEV_SEQ_CONF=disk0:;disk1:
  BOOT_DEV_SEQ_OPER=disk0:;disk1:
```

In the following example, the **show system backup** command is entered with the **details** keyword to display additional information on the configuration and software package files stored on the backup device. Because this command is entered in administration EXEC mode, the backup information for both the administration and SDR configurations is displayed.

```
RP/0/RP0/CPU0:router(admin)# show system backup details

System Backup information for node0_0_CPU0 on disk1:
=====================================================
  Last Backup Successful
  Backup started at Sat Jun 24 12:22:10 2009
   ended at Sat Jun 24 12:42:11 2009
  Verify started at Sat Jun 24 12:42:12 2009
   ended at Sat Jun 24 12:48:47 2009
  BOOT_DEV_SEQ_CONF=disk0:;disk1:
  BOOT_DEV_SEQ_OPER=disk0:;disk1:
  Admin configuration last commit record on disk1:
    Device          Commitid    Time Stamp
    disk1:          2000000010  23:07:59 UTC Fri Jun 09 2009

  SDR configuration last commit record on disk1:
    Device          Commitid    Time Stamp
    disk1:          1000000030  11:56:43 UTC Thu Jun 22 2006

  Active software packages on disk1:
  hfr-os-mbi-3.7.2
  hfr-base-3.7.2
  hfr-admin-3.7.2
  hfr-fwdg-3.7.2
  hfr-lc-3.7.2
  hfr-rout-3.7.2
  hfr-diags-3.7.2
  hfr-k9sec-3.7.2
  hfr-mcast-3.7.2
  hfr-mgbl-3.7.2
  hfr-mpls-3.7.2
  No Inactive software packages on disk1:
```

In the following example, backup information is displayed for backups located on disk1: in all RPs in the system. In this example, a separate backup was created on disk1: of node 0/3/CPU0 for a non-owner SDR.

```
RP/0/RP0/CPU0:router(admin)# show system backup disk1: location all

System Backup information for node0_0_CPU0 on disk1:
=====================================================
  Last Backup Successful
  Backup started at Sat Jun 24 12:22:10 2006
   ended at Sat Jun 24 12:42:11 2006
  Verify started at Sat Jun 24 12:42:12 2006
   ended at Sat Jun 24 12:48:47 2006
  BOOT_DEV_SEQ_CONF=disk0:;disk1:
  BOOT_DEV_SEQ_OPER=disk0:;disk1:
```

```
System Backup information for node0_3_CPU0 on disk1:
=====================================================
Last Backup Successful
Backup started at Sat Jun 24 13:02:23 2006
 ended at Sat Jun 24 13:21:30 2006
Verify started at Sat Jun 24 13:21:30 2006
 ended at Sat Jun 24 13:27:55 2006
BOOT_DEV_SEQ_CONF=disk0:;disk1:
BOOT_DEV_SEQ_OPER=disk0:;disk1:
```

Configuration Examples for Enabling Disk Mirroring

Enabling Disk Mirroring: Example

In the following example, disk mirroring is enabled on a router:

```
format disk1: partition

This operation will destroy all data on "disk1:" and partition device.
Continue? [confirm] y

Device partition disk1: is now formated and is available for use.

configure
  mirror location 0/0/cpu0 disk0:disk1:
  commit
```

show mirror Command Output: Example

```
RP/0/RP0/CPU0:router# show mirror location all

Tue Dec  7 05:58:11.187 PST

Mirror Information for 0/4/CPU0.
=====================================================
 Mirroring Enabled
   Configured Primary:         disk0:
   Configured Secondary:       disk1:

 Current Mirroring State:      Redundant
   Current Physical Primary:   disk0:
   Current Physical Secondary: disk1:

 Mirroring Logical Device:     disk0:
 Mirroring Logical Device2:    disk1:

 Physical Device     State       Flags
-----------------------------------------------------
  disk0:           Available     Enabled
  disk1:           Available     Enabled
  compactflash:    Not Present
  harddiska:       Not Present
  disk0a:          Available
```

```
  disk1a:          Available
  compactflasha:   Not Present
  harddisk:        Available

Mirroring Rommon Variable
 BOOT_DEV_SEQ_CONF = disk0:;disk1:
 BOOT_DEV_SEQ_OPER = disk0:;disk1:
 MIRROR_ENABLE = Y


Mirror Information for 0/4/CPU1.
=========================================================
 Mirroring Enabled
   Configured Primary:         disk0:
   Configured Secondary:       disk1:

 Current Mirroring State:      Redundant
   Current Physical Primary:   disk0:
   Current Physical Secondary: disk1:

 Mirroring Logical Device:     disk0:
 Mirroring Logical Device2:    disk1:

 Physical Device     State       Flags
---------------------------------------------------------
  disk0:           Available     Enabled
  disk1:           Available     Enabled
  compactflash:    Not Present
  harddiska:       Not Present
  disk0a:          Available
  disk1a:          Available
  compactflasha:   Not Present
  harddisk:        Available

Mirroring Rommon Variable
 BOOT_DEV_SEQ_CONF = disk0:;disk1:
 BOOT_DEV_SEQ_OPER = disk0:;disk1:
 MIRROR_ENABLE = Y


Mirror Information for 0/RP0/CPU0.
=========================================================
 Mirroring Enabled
   Configured Primary:         disk0:
   Configured Secondary:       disk1:

 Current Mirroring State:      Redundant
 Current Physical Primary:   disk0:
   Current Physical Secondary: disk1:

 Mirroring Logical Device:     disk0:
 Mirroring Logical Device2:    disk1:

 Physical Device     State       Flags
---------------------------------------------------------
  disk0:           Available     Enabled
  disk1:           Available     Enabled
  compactflash:    Not Present
  harddiska:       Not Present
  disk0a:          Available
  disk1a:          Available
  compactflasha:   Not Present
  harddisk:        Available
```

```
Mirroring Rommon Variable
 BOOT_DEV_SEQ_CONF = disk0:;disk1:
 BOOT_DEV_SEQ_OPER = disk0:;disk1:
 MIRROR_ENABLE = Y


Mirror Information for 0/RP1/CPU0.
=========================================================
 Mirroring Enabled
   Configured Primary:         disk0:
   Configured Secondary:       disk1:

 Current Mirroring State:      Redundant
   Current Physical Primary:   disk0:
   Current Physical Secondary: disk1:

 Mirroring Logical Device:     disk0:
 Mirroring Logical Device2:    disk1:

 Physical Device     State       Flags
---------------------------------------------------------
  disk0:           Available     Enabled
  disk1:           Available     Enabled
  compactflash:    Not Present
  harddiska:       Not Present
  disk0a:          Available
  disk1a:          Available
  compactflasha:   Not Present
  harddisk:        Available

Mirroring Rommon Variable
 BOOT_DEV_SEQ_CONF = disk0:;disk1:
 BOOT_DEV_SEQ_OPER = disk0:;disk1:
 MIRROR_ENABLE = Y
```

show mirror Command Output on a Multishelf System: Example

```
RP/0/RP0/CPU0:router(admin)# show mirror location F0/SC0/CPU0

Mirror Information for F0/SC0/CPU0.
=========================================================
 Mirroring Disabled

 Current Mirroring State:      Not Configured
   Current Physical Primary:   disk0:
   Current Physical Secondary: Not Set

 Mirroring Logical Device:     disk0:

 Physical Device     State       Flags
---------------------------------------------------------
  disk0:           Available     Enabled
  disk1:           Available     Formatted
  compactflash:    Not Present
  disk0a:          Available
  disk1a:          Available     Formatted
  compactflasha:   Not Present

Mirroring Rommon Variable
 BOOT_DEV_SEQ_CONF =
```

```
 BOOT_DEV_SEQ_OPER =
 MIRROR_ENABLE =
```

mirror verify Command Output: Example

```
RP/0/RP0/CPU0:router# mirror verify

Mirror Verify Information for 0/0/CPU0.
=======================================================
  Primary device and secondary device are fully synchronized.
```

Additional References

The following sections provide references related to disk backup and disk mirroring configuration.

Related Documents

Related Topic	Document Title
Initial system bootup and configuration information for a router using the Cisco IOS XR software	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*
Information about user groups and task IDs	*Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*
Cisco IOS XR command master list	*Cisco IOS XR Commands Master List for the Cisco CRS Router*
Cisco IOS XR boot commands	*Boot Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

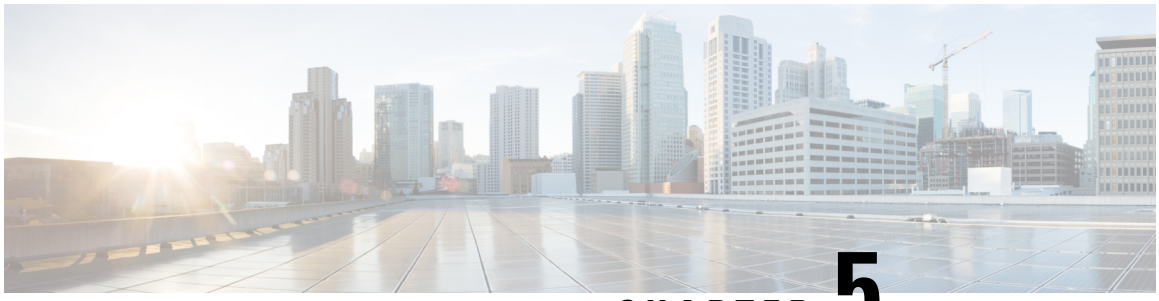
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 5

Software Entitlement

Cisco IOS XR software contains all the supported features for a given release. Before the introduction of software entitlement on Cisco IOS XR software, you could freely activate all available software packages on your network devices and could enable all the bundled features. The same was true for hardware, as well. For example, modular services cards (MSCs) can be programmed to run in 20-Gbps or 40-Gbps modes. Before Release 3.5.0 of Cisco IOS XR software, all MSCs ran in 40-Gbps mode. Software entitlement has been introduced so you pay only for the features that you need today, but can upgrade when necessary while keeping your investment safe. Licensing enables you to purchase individual software features and upgrade hardware capacity in a safe and reliable way.

To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 11: Feature History for Software Entitlement

Release	Modification
Release 3.5.0	The software entitlement feature was introduced.
Release 4.3.0	CRS- FP140 license was supported.

This model contains the following topics:

What Is Software Entitlement?

Software entitlement is a system that consists of a license manager on a Cisco IOS XR device that manages licenses for various software and hardware features. The license manager parses and authenticates a license before accepting it. The software features on the router use the license manager APIs to check out and release licenses. Licenses are stored in persistent storage on the router.

Core routing features are available for use without any license. The following features can be enabled on your router using licenses:

Layer 3 VPN

Layer 3 (virtual private network) VPN can be configured only if there is an available Layer 3 VPN license for the modular services card (MSC) slot on which the feature is being configured. The Layer 3 VPN

license is included with the MSC, but it must be purchased separately to use with the 40G forwarding processor (CRS-FP40).

See the following modules in *MPLS Configuration Guide for the Cisco CRS Routers* for information about Layer 3 VPN configurations:

- *Implementing MPLS Layer 3 VPNs on the Cisco IOS XR Software*
- *Implementing Virtual Private LAN Services on the Cisco IOS XR Software*

Performance Netflow

The CRS-FP40 card can support down to 1:360 netflow sample rate running 45 Mpps at a 40-Gbps line rate. Without this license, the CRS-FP40 and CRS-FP140 can support down to 1:1500 netflow sample rate. The MSC includes this license.

Advanced Features

Lawful Interception, GRE tunnel, and L2TPv3. The MSC includes this license.

Modular Services Card Bandwidth

Modular services cards (MSCs) can operate at 40-Gbps throughput when a license is enabled. You must use the **hw-module linecard throughput** command to enable 40-Gbps throughput once a license is available.

Scale

The CRS-FP140 can support increased route scale (more than 1 million IPv4 routes) and traffic engineering scale (more than 3,000 tunnels per system using licenses.

Multichassis Support

The CRS-FP140 can support a multichassis system using a license.

Implementing Default Licensing

Prerequisites for Configuring Software Entitlement

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Information About Default (Traditional) Licensing

To configure software license entitlements using the default mode of licensing, you need to understand the concepts described in this module.

Types of Licenses

The following types of licenses are currently defined:

- Permanent licenses—Licenses that enable a designated feature permanently, as long as the license resides on the router.

- Implicit evaluation licenses—Set of evaluation licenses that are included with the software image (upgrade or initial install). Like regular evaluation licenses, these are valid for a period of ninety days, but the countdown to expiry starts as soon as the router is booted with an image containing these licenses.

SDR License Pools

License pools are maintained according to secure domain router (SDR). By default, all added licenses are allocated to the owner SDR license pool, and they can be freely allocated to any slot in any SDR. Features on cards belonging to the owner SDR are granted licenses based on availability in the owner SDR license pool.

You can create SDR-specific license pools by using the **license pool create** command. License requests for features running on cards belonging to SDRs with SDR-specific pools are not served from the owner license pool, even if the owner SDR license pool has licenses available. You must allocate licenses from the owner SDR to other SDRs explicitly for these license requests to succeed. Similarly, if a slot in the owner SDR had a feature license and subsequently was moved to some other SDR with an SDR-specific license pool, the feature license stays with the original SDR license pool.

You can explicitly add new licenses to a particular SDR license pool or move available licenses from the owner SDR license pool to any other SDR.

Chassis-Locked Licenses

Licenses are locked to a unique device identifier (UDI). The UDI is comprised of the chassis serial number, along with an additional identifier. The complete set of UDI information can be displayed using the **show license udi** command. The license manager parses the user-provided license and verifies that it is valid for the chassis it is running on and determines if the license is being readded.

Multiple single-chassis Cisco CRS-1 routers can be connected using a fabric chassis to form a multishelf system. In multishelf systems, licenses are locked to the UDI of the individual chassis, but can be used to enable features on any chassis. The list of available licenses for a multishelf system is a collection of all licenses on each contributing chassis.

Slot-Based Licenses

Feature licenses are allocated to router slots and not cards. Therefore, if a card is replaced, the existing license is applied to the newly inserted card. For example, if you have eight licenses for Layer 3 VPN in the system, you can configure Layer 3 VPN features on any eight cards in an SDR , and the licenses are allocated to the slots within which the cards are installed. If a card is removed from one of these licensed slots, say slot 3, and entered into an empty slot with no license, say slot 5, the license remains with slot 3 and the feature cannot be activated on slot 5 with the permanent license entered earlier by the user. In this case, you can release the license to the appropriate license pool by removing the configuration of the card (while it is inserted), or by using the **license move slot** command. When you configure the feature on slot 5, the license is checked out.

Using Implicit Licenses After a Software Image Upgrade

When you upgrade your Cisco IOS XR software image from a release that does not support software entitlement to one that does, you are provided with implicit licenses to use for all configured features in your original configuration. This enables you to upgrade your software without worrying about the implications of software entitlement.

Implicit licenses are good for a period of 90 days. As soon as the new image boots, the license manager displays a syslog message to the console once a day, indicating that an implicit license is being used and must

be replaced with a permanent license. This frequency increases to once an hour on the last day before the expiry of the implicit licenses, to ensure that you do not miss it.

Before your implicit licenses expire, you should purchase licenses for all features that you want to keep running.

Features that Require Licenses After a Software Image Upgrade

When you upgrade your Cisco IOS XR software image from a release that does not support software entitlement to one that does, a warning message is displayed to the console port for each feature that requires a license. You must acquire either an evaluation license or a permanent license in order to continue using any features that require a license.

During an install activate operation, if the installation fails to acquire a license (through the license manager) for a package that requires licensing then the install operation is allowed but a warning message similar to the following is displayed:

```
Fri Nov 20 15:26:52.311 UTC

Install operation 3 started by user 'lab' via CLI at 15:21:18
UTC Fri Nov 20 2009.
(admin) install activate disk0:hfr-mcast-p-4.0.0.3P disk0:hfr-mgbl-p-4.0.0.3P
Install operation 3 completed successfully at 15:25:21 UTC Fri Nov 20 2009.

Install logs:
    Install operation 3 '(admin) install activate disk0:hfr-mcast-p-4.0.0.3P
      disk0:hfr-mgbl-p-4.0.0.3P' started by user 'lab' via CLI at 15:21:18 UTC
      Fri Nov 20 2009.
    Warning:  There is no valid license for the following packages:
    Warning:
    Warning:      disk0:hfr-mcast-supp-4.0.0.3P
    Warning:      disk0:hfr-mgbl-supp-4.0.0.3P
    Warning:
    Info:     The following sequence of sub-operations has been determined to
    Info:     minimize any impact:
    Info:
    Info:     Sub-operation 1:
    Info:         Install Method: Parallel Process Restart
    Info:         hfr-mcast-supp-4.0.0.3P
    Info:         iosxr-mcast-4.0.0.3P
    Info:
    Info:     Sub-operation 2:
    Info:         Install Method: Parallel Process Restart
    Info:         hfr-mgbl-supp-4.0.0.3P
    Info:         iosxr-mgbl-4.0.0.3P
    Info:
    Info:     The changes made to software configurations will not be
    Info:     persistent across system reloads. Use the command '(admin)
    Info:     install commit' to make changes persistent.
    Info:     Please verify that the system is consistent following the
    Info:     software change using the following commands:
    Info:         show system verify
    Info:         install verify packages
    Install operation 3 completed successfully at 15:25:21 UTC Fri Nov 20 2009.
```

If you activate an SMU whose corresponding package requires a license but a license was not acquired successfully, then the install operation is allowed but a warning message similar to the following is displayed:

```
Wed Nov 25 15:02:23.418 PST
```

```
Install operation 8 started by user 'lab' via CLI at 14:59:46 PST Wed Nov 25 2009.
(admin) install activate id 7
Install operation 8 completed successfully at 15:02:13 PST Wed Nov 25 2009.

Install logs:
    Install operation 8 '(admin) install activate id 7' started by user 'lab'
      via CLI at 14:59:46 PST Wed Nov 25 2009.
    Info:     This operation will activate the following packages:
    Info:         disk0:comp-crs1-4.0.0.3P.CSCee40001-1.0.0
    Info:         disk0:comp-crs1-4.0.0.3P.CSCee30001-1.0.0
    Info:         disk0:comp-crs1-4.0.0.3P.CSCee20001-1.0.0
    Info:         disk0:comp-crs1-4.0.0.3P.CSCee10001-1.0.0
    Info:     The following SMUs are not being activated as they do not apply to
    Info:     any packages on the router:
    Info:
    Info:         disk0:hfr-diags-supp-4.0.0.3P.CSCee30001-1.0.0
    Info:         disk0:hfr-fpd-4.0.0.3P.CSCee40001-1.0.0
    Info:
    Warning:  There is no valid license found for package 'disk0:hfr-mcast-supp-4.0.0.3P'
    Warning:  when activating SMU 'disk0:hfr-mcast-supp-4.0.0.3P.CSCee10001-1.0.0'.
    Warning:  There is no valid license found for package 'disk0:hfr-mgbl-supp-4.0.0.3P'
    Warning:  when activating SMU 'disk0:hfr-mgbl-supp-4.0.0.3P.CSCee20001-1.0.0'.
    Warning:
    Info:     The following sequence of sub-operations has been determined to minimize any
    Info:     impact:
    Info:     Sub-operation 1:
    Info:         Install Method: Parallel Process Restart
    Info:         hfr-mcast-supp-4.0.0.3P.CSCee10001-1.0.0
    Info:
    Info:     Sub-operation 2:
    Info:         Install Method: Parallel Process Restart
    Info:         hfr-mgbl-supp-4.0.0.3P.CSCee20001-1.0.0
    Info:
    Info:     The changes made to software configurations will not be persistent
    Info:     across system reloads. Use the command '(admin) install commit' to
    Info:     make changes persistent.
    Info:     Please verify that the system is consistent following the software
    Info:     change using the following commands:
    Info:         show system verify
    Info:         install verify packages
    Install operation 8 completed successfully at 15:02:13 PST Wed Nov 25 2009.
```

Configure Licenses Using Default Licensing

Adding a License for a New Feature

This task describes how to acquire a permanent license for a feature that you have purchased or an evaluation license for a feature that you have arranged with your sales representative to try. Use this procedure to replace implicit or evaluation licenses with permanent licenses.

Note Evaluation licenses cannot be installed if permanent licenses for the same feature are valid on the chassis. Also note that if you add a permanent license to a chassis, all evaluation or implicit licenses of the same type are disabled.

Before you begin

You must have purchased the feature for which you are adding the license. When you purchase the feature, you are provided with a product authorization key (PAK) that you use to download the license.

Note All implicit or evaluation licenses for a feature are disabled when at least one permanent license for a feature is added to the router. This is true even if you had more evaluation licenses than permanent licenses.

SUMMARY STEPS

1. **admin**
2. **show license udi**
3. http://www.cisco.com/go/license
4. Copy the license to your TFTP server.
5. **admin**
6. **license add** *license-name* [**sdr** *sdr-name*]
7. **configure**
8. **license** *license-name* **location** {**all** | *node-id*}
9. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

Step 2	**show license udi**
	Example:
	`RP/0/RP0/CPU0:P1_CRS-8(admin)# show license udi`
	`Tue Jul 14 03:22:21.829 PST`
	`Local Chassis UDI Information:`
	`  PID         : CRS-8-LCC`
	`  S/N         : TBA09370035`
	`  Operation ID: 0`
		Displays the UDI of the chassis. This consists of a product identifier (PID), serial number (S/N), and operation identifier (Operation ID).

Step 3	http://www.cisco.com/go/license	Go to the license tool on Cisco.com. You must log in to the site before you can access the license tool. Follow the instructions for product license registration. You are required to enter the feature PAK and the chassis UDI to acquire the license.

	Command or Action	Purpose
		Note If you are installing a permanent license, you should have received the PAK when you purchased the feature. If you are installing an evaluation license, your sales representative should provide you with the PAK.
Step 4	Copy the license to your TFTP server.	You will be issued a license. You can copy the license and store it on your computer, or alternatively, you can request that the license be sent to you in an e-mail. When you have received the license, copy it to a TFTP server that is accessible by your router.
Step 5	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 6	**license add** *license-name* [**sdr** *sdr-name*] **Example:** `RP/0/RP0/CPU0:router(admin)# license add tftp://192.10.10.10/mylicenses/lc40g_lic`	Adds the license to the SDR license pool. By default, the license is added to the owner SDR license pool.
Step 7	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 8	**license** *license-name* **location** {**all** | *node-id*} **Example:** `RP/0/RSP0/CPU0:router(admin-config)# license A9K-ADV-OPTIC-LIC location 0/0/CPU0`	Binds the license to the slot where it is to be used.
Step 9	**exit** **Example:** `RP/0/RP0/CPU0:router(admin)# exit`	Exits administration EXEC mode.

What to do next

To use the feature associated with the added license, you must configure it on your router. To configure Layer 3 VPN, see the *Implementing MPLS Layer 3 VPNs on Cisco IOS XR Software* module in *MPLS Configuration Guide for the Cisco CRS Routers*.

To verify that your MSC is operating at 40-Gbps throughput, use the **show hw-module linecard throughput** command. To verify that your Layer 3 VPN configuration is operational, use the **show rsi interface all global** command.

Enabling 40-Gbps Throughput on an MSC

To configure a modular services card (MSC) to operate at 40-Gbps throughput, perform the following task. This must be performed when you add permanent licenses to your router to replace implicit licenses. It also must be performed if you are purchasing a new license for 40-Gbps throughput after using the MSC with the default 20-Gbps throughput.

Note When you upgrade your image from an image that does not support software entitlement to one that does, all existing MSCs default to the 40-Gbps configuration as long as there are active implicit licenses. When the implicit licenses expire, the configuration reverts to the default 20-Gbps.

Before you begin

You must have a license on your system for 40-Gbps throughput to enable this feature. This could be an implicit license, evaluation license, or permanent license.

SUMMARY STEPS

1. **configure**
2. **hw-module linecard throughput 40 location** *node-id*
3. **commit**
4. **show hw-module linecard throughput**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**hw-module linecard throughput 40 location** *node-id* **Example:** `RP/0/RP0/CPU0:router(config)# hw-module linecard throughput 40 location 0/6/0`	Enables 40-Gbps throughput on the MSC in the specified *node-id* . If this command is not added to your configuration, the MSC continues to work at 20 Gbps, even if you have a valid license.
Step 3	**commit**	
Step 4	**show hw-module linecard throughput** **Example:** `RP/0/RP0/CPU0:router# show hw-module linecard throughput`	Displays the operational throughput on the MSCs in the router.

Examples

The following example shows sample output from the **show hw-module linecard throughput** command. In this example, the MSC in slot 6 has no license to run at 40 Gbps, while the MSC in slot 1 has a license and is operating at 40 Gbps.

```
RP/0/RP0/CPU0:router# show hw-module linecard throughput

   -------------------- Throughput --------------------
   Location        Configured      Lic Acquired    Operating

   0/6/CPU0         no config          No            20G

   0/1/CPU0         no config          Yes           40G
```

Backing Up Licenses

When your router is configured with the licenses that you require, you should perform this task to back up all licenses. Backing up licenses makes it easier to restore them if there is a problem.

SUMMARY STEPS

1. **admin**
2. **license backup** *backup-file*
3. **show license backup** *backup-file*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

Step 2	**license backup** *backup-file*
	Example:
	`RP/0/RP0/CPU0:router(admin)# license backup disk1:/license_back`
	`License command "license backup disk1:/license_back" completed successfully.`
		Backs up all licenses on the router to a backup file in the specified location. The backup file can be a local file or a remote file on a TFTP or RCP server.

Step 3	**show license backup** *backup-file* **Example:** `RP/0/RP0/CPU0:router(admin)# show license backup disk1:/license_back`	Displays the contents of the backup file.

Examples

The following example shows sample output from the **show license backup** command.

```
RP/0/RP0/CPU0:router(admin)# show license backup disk1:/license_back
```

```
Local Chassis UDI Information:
  S/N         : TBA09370035
  Operation ID:   5
Licenses      :
FeatureID    Type                      #installed
CRS-MSC-40G  Slot based,    Permanent          2
XC-L3VPN     Slot based,    Permanent          1
```

Restoring Licenses

If your licenses become corrupted, and you have previously created a backup of your licenses, you can perform this task to restore the licenses to your router.

Before you begin

You must have created a backup file of your licenses before you can restore them on your router.

SUMMARY STEPS

1. **admin**
2. **show license backup** *backup-file*
3. **license restore** *backup-file*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**show license backup** *backup-file* **Example:** `RP/0/RP0/CPU0:router(admin)# show license backup disk1:/license_back`	Displays the contents of the backup file. You should verify the contents of the backup file before you restore your licenses.
Step 3	**license restore** *backup-file* **Example:** `RP/0/RP0/CPU0:router(admin)# license restore disk1:/license_back`	Restores all licenses on the router from a backup file in the specified location. This can be a local file, or a remote file on a TFTP or RCP server.

Examples

This example shows sample output from the **license restore** command.

```
RP/0/RP0/CPU0:router(admin)# license restore disk1:/license_back

Info: This command will erase all existing licenses.
Info: It is strongly recommended to backup existing licenses first.
Do you wish to proceed? [yes/no]: y
```

```
License command "license restore disk1:/license_back" completed successfully.
```

Transferring Licenses to a new Route Switch Processor 440

To upgrade a route switch processor (RSP) to the RSP440 with your active licenses, perform this task.

SUMMARY STEPS

1. Save the original license file that you received from Cisco, in a USB drive or TFTP server.
2. **save configuration running**
3. Replace the RSP with the RSP440.
4. **load**
5. Add all licenses as described in Adding a License for a New Feature, on page 107.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Save the original license file that you received from Cisco, in a USB drive or TFTP server.	Stores the license files in a location that is accessible to the new RSP after it is installed.

Step 2	**save configuration running**
	Example:
	`RP/0/RP0/CPU0:router# configure`
	`RP/0/RP0/CPU0:router(config)# save configuration running tftp://192.10.10.10/mylicenses/rc_03132013`
	`RP/0/RP0/CPU0:router(config)# exit`
	`RP/0/RP0/CPU0:router# admin`
	`RP/0/RP0/CPU0:router(admin)# configure`
	`RP/0/RP0/CPU0:router(admin-config)# save configuration running tftp://192.10.10.10/mylicenses/rc_admin_03132013`
		Saves the running-configuration from both global configuration and administration configuration modes to files. The administration configuration contains user group information that is required for the licenses.

Step 3	Replace the RSP with the RSP440.	

Step 4	**load**
	Example:
	`RP/0/RP0/CPU0:router# configure`
	`RP/0/RP0/CPU0:router(config)# load tftp://192.10.10.10/mylicenses/rc_03132013`
	`RP/0/RP0/CPU0:router(config)# exit`
	`RP/0/RP0/CPU0:router# admin`
	`RP/0/RP0/CPU0:router(admin)# configure`
	`RP/0/RP0/CPU0:router(admin-config)# load tftp://192.10.10.10/mylicenses/rc_admin_03132013`
		Loads the saved running-configuration files on the new RSP440. This must be done in both global configuration mode and administration configuration mode.

Step 5	Add all licenses as described in Adding a License for a New Feature, on page 107.	Installs the licenses to the new RSP.

Upgrading Line Cards and Licenses

You may have a number of licenses running on the line cards in your router. Before you upgrade these line cards to advanced models, you need to install new licenses that are appropriate for these line cards. This task describes the steps necessary to upgrade your line cards and their licenses.

Before you begin

- You must have purchased all relevant licenses for the line cards that you are upgrading, prior to performing this task.
- You must have placed the license files on your router disk drive or a TFTP server such that they accessible from your router. Refer to Adding a License for a New Feature, on page 107 for more information.

SUMMARY STEPS

1. **admin**
2. **license add** *license-name*
3. **configure**
4. Remove the old line cards and install the new ones.
5. **show license**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**license add** *license-name* **Example:** `RP/0/RP0/CPU0:router(admin)# license add tftp://192.10.10.10/mylicenses/A9K-24X10G-AIP-TR-lic`	Adds the license to the SDR license pool. By default, the license is added to the owner SDR license pool.
Step 3	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 4	Remove the old line cards and install the new ones.	Associates the new license on the slot with the new line card. **Note** You need to wait for the new line cards to boot.
Step 5	**show license** **Example:** `RP/0/RP0/CPU0:router# show license`	Verifies the status of the licenses. **Note** Old licenses remaining on the router are not a problem.

	Command or Action	Purpose

	Thu May 31 08:52:39.876 PST
	
	FeatureID: A9K-24X10G-AIP-TR (Slot based, Permanent)
	  Total licenses 1
	  Available for use 0
	  Allocated to location 0
	  Active 1
	  Store name Permanent
	  Store index 3
	  Pool: Owner
	  Total licenses in pool: 1
	  Status: Available 0 Operational: 1
	  Locations with licenses: (Active/Allocated) [SDR]
	  0/0/CPU0 (1/0) [Owner]	

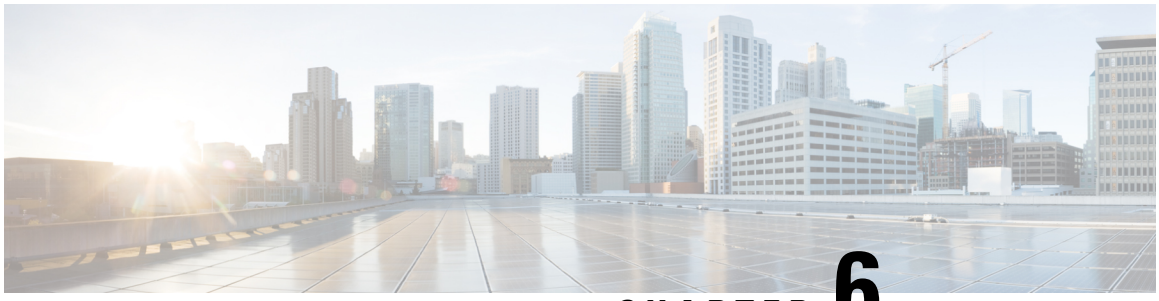
Troubleshooting License Issues after a Software Upgrade

In the instance that you were running Cisco IOS XR Release 3.9.0 and had the optic feature enabled on a interface and the A9K-ADV-OPTIC-LIC license was active on a particular slot, when you upgrade to Cisco IOS XR Release 4.0.0, the A9K-ADV-OPTIC-LIC license is still active, but you may get the following warning message:

```
RP/0/RSP0/CPU0:Jul 27 14:22:22.594 : licmgr[236]:
%LICENSE-LICMGR-4-PACKAGE_LOCATION_LICENSE_INVALID :
Feature associated to package A9K-ADV-OPTIC-LIC configured
on node 0/4/CPU0 without a valid license
```

To solve this issue, configure the **license** command in administration EXEC mode. This binds the A9K-ADV-OPTIC-LIC license to the slot on which you are using the license. For example:

```
RP/0/RSP0/CPU0:router(admin-config)# license A9K-ADV-OPTIC-LIC location 0/4/CPU0
RP/0/RSP0/CPU0:router(admin-config)# commit
```



CHAPTER 6

Managing Router Hardware

This chapter describes the command-line interface (CLI) techniques and commands used to manage and configure the hardware components of a router running the Cisco IOS XR software.

For complete descriptions of the commands listed in this module, see Additional References, on page 175. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 12: Feature History for Managing Router Hardware with Cisco IOS XR Software

Release	Modification
Release 2.0	This feature was introduced.
Release 3.2	Logical router (LR) was first supported.
Release 3.3.0	The term logical router (LR) was changed to secure domain router (SDR).
Release 3.5.0	Flash disk recovery was implemented.

This module contains the following topics:

Prerequisites for Managing Router Hardware

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Displaying Hardware Status

This section describes how to display different types of hardware status information.

Displaying SDR Hardware Version Information

To display hardware version information for the components assigned to a secure domain router (SDR), connect to the appropriate designated secure domain router shelf controller (DSDRSC) and enter the **show diag** command in EXEC mode. The displayed information includes the card serial number and the ROMMON software version.

The syntax for the **show diag** command in EXEC mode is:

show diag [*node-id* | **details** | **summary**]

In the following example, the **show diag** command displays information for all nodes in the SDR:

```
RP/0/RP0/CPU0:router# show diag

  PLIM 0/1/CPU0 : JACKET CARD
    MAIN:  board type 580070
           800-23819-03 rev C0
           dev N/A
           S/N SAD094401CR
    PCA:   73-8982-06 rev C0
    PID:   CRS1-SIP-800
    VID:   V01
    CLEI:  COUIAAMCAA
    ECI:   134912
    Board State : IOS XR RUN
    PLD:   Motherboard: 0x0025, Processor: 0xda13, Power: N/A
    MONLIB: QNXFFS Monlib Version 3.0
    ROMMON: Version 1.40(20050525:193559)  [CRS-1 ROMMON]
    Interface port config:  0 Ports
    Optical reach type:  Unknown
    Connector type:  MT-P

  NODE 0/1/0 : 4xOC3 POS SPA
    MAIN:  board type 0440
           68-2169-01 rev C0
           dev N/A
           S/N JAB093309PA
    PCA:   73-9313-04 rev B0
    PID:   SPA-4XOC3-POS
    VID:   V01
    CLEI:  IPUIAFNRAA

  NODE 0/1/5 : 8xGE SPA
```

```
  MAIN:  board type 044f
         68-2239-01 rev A0
         dev N/A
         S/N SAD0937022J
  PCA:   73-8557-03 rev A0
  PID:   SPA-8X1GE
  VID:   V01
  CLEI:  CNUIAH6AAA

PLIM 0/6/CPU0 : JACKET CARD
  MAIN:  board type 580070
         800-23819-03 rev C0
         dev N/A
         S/N SAD094203W2
  PCA:   73-8982-06 rev C0
  PID:   CRS1-SIP-800
  VID:   V01
  CLEI:  COUIAAMCAA
  ECI:   134912
  Board State : IOS XR RUN
  PLD:   Motherboard: 0x0025, Processor: 0xda13, Power: N/A
  MONLIB: QNXFFS Monlib Version 3.0
  ROMMON: Version 1.40(20050525:193559)  [CRS-1 ROMMON]
  Interface port config:  0 Ports
  Optical reach type:  Unknown
  Connector type:  MT-P

NODE 0/6/0 : 4xOC3 POS SPA
  MAIN:  board type 0440
         68-2169-01 rev C0
         dev N/A
         S/N JAB093309MG
  PCA:   73-9313-04 rev B0
  PID:   SPA-4XOC3-POS
  VID:   V01
  CLEI:  IPUIAFNRAA

NODE 0/6/4 : 8xOC3/OC12 POS SPA
  MAIN:  board type 0404
         68-2164-01 rev 34
         dev N/A
         S/N JAB094706L9
  PCA:   73-9941-02 rev 04
  PID:   SPA-8XOC12-POS
  VID:   V01
  CLEI:  SOUIAA8BAA

NODE 0/6/5 : 8xGE SPA
  MAIN:  board type 044f
         68-2239-01 rev A0
         dev N/A
         S/N SAD093909GM
  PCA:   73-8557-03 rev A0
  PID:   SPA-8X1GE
  VID:   V01
  CLEI:  CNUIAH6AAA

NODE 0/RP0/CPU0 : RP
  MAIN:  board type 100002
         800-22921-10 rev B0
         dev 080366, 080181
         S/N SAD093507J8
  PCA:   73-8564-10 rev B0
  PID:   CRS-8-RP
```

```
    VID:   V01
    CLEI:  IPUCABWBAA
    ECI:   129507
    Board State : IOS XR RUN
    PLD:   Motherboard: 0x0038, Processor: 0x0038, Power: 0x0000
    MONLIB: QNXFFS Monlib Version 3.0
    ROMMON: Version 1.40(20050525:193559)  [CRS-1 ROMMON]

  NODE 0/RP1/CPU0 : RP
    MAIN:  board type 100002
           800-22921-10 rev B0
           dev 080366, 080181
           S/N SAD093507JP
    PCA:   73-8564-10 rev B0
    PID:   CRS-8-RP
    VID:   V01
    CLEI:  IPUCABWBAA
    ECI:   129507
    Board State : IOS XR RUN
    PLD:   Motherboard: 0x0038, Processor: 0x0038, Power: 0x0000
    MONLIB: QNXFFS Monlib Version 3.0
    ROMMON: Version 1.40(20050525:193559)  [CRS-1 ROMMON]
```

In the following example, the **show diag** command displays information for a single node:

```
RP/0/RP0/CPU0:router# show diag 0/RP0/CPU0

  NODE 0/RP0/CPU0 : RP
    MAIN:  board type 100002
           800-22921-10 rev B0
           dev 080366, 080181
           S/N SAD093507J8
    PCA:   73-8564-10 rev B0
    PID:   CRS-8-RP
    VID:   V01
    CLEI:  IPUCABWBAA
    ECI:   129507
    Board State : IOS XR RUN
    PLD:   Motherboard: 0x0038, Processor: 0x0038, Power: 0x0000
    MONLIB: QNXFFS Monlib Version 3.0
    ROMMON: Version 1.40(20050525:193559)  [CRS-1 ROMMON]
```

Displaying System Hardware Version Information

To display hardware version information for all or some of the components assigned in a system, connect to the designated shelf controller (DSC) and enter the **show diag** command in administration EXEC mode. When this command is entered in administration EXEC mode, you can display information on RPs, MSCs or line cards, fabric cards, and system components such as the chassis, fan trays, and power supplies.

Note If you enter the **show diag** command in EXEC mode, the software displays only the hardware assigned to the SDR to which you are connected.

The syntax for the **show diag** command in administration EXEC mode is:

show diag [*node-id* | **chassis** | **details** | **fans** | **memory** | **power-supply** | **summary**]

Tip For information on the software version, use the **show version** command.

In the following example, the **show diag** command displays information for all nodes in the system:

```
RP/0/RP0/CPU0:router(admin)# show diag

  NODE 0/1/SP : MSC(SP)
    MAIN:  board type 500060
           800-25021-05 rev B0
           dev 079239
           S/N SAD09280BS9
    PCA:   73-7648-08 rev B0
    PID:   CRS-MSC
    VID:   V02
    CLEI:  IPUCAC1BAA
    ECI:   132502
    Board State : IOS XR RUN
    PLD:   Motherboard: 0x0025, Processor: 0xda13, Power: N/A
    MONLIB: QNXFFS Monlib Version 3.0
    ROMMON: Version 1.40(20050525:193402)  [CRS-1 ROMMON]

  PLIM 0/1/CPU0 : JACKET CARD
    MAIN:  board type 580070
           800-23819-03 rev C0
           dev N/A
           S/N SAD094401CR
    PCA:   73-8982-06 rev C0
    PID:   CRS1-SIP-800
    VID:   V01
    CLEI:  COUIAAMCAA
    ECI:   134912
    Board State : IOS XR RUN
    PLD:   Motherboard: 0x0025, Processor: 0xda13, Power: N/A
    MONLIB: QNXFFS Monlib Version 3.0
    ROMMON: Version 1.40(20050525:193559)  [CRS-1 ROMMON]
    Interface port config:  0 Ports
    Optical reach type:  Unknown
    Connector type:  MT-P

  NODE 0/1/0 : 4xOC3 POS SPA
    MAIN:  board type 0440
           68-2169-01 rev C0
           dev N/A
           S/N JAB093309PA
    PCA:   73-9313-04 rev B0
    PID:   SPA-4XOC3-POS
    VID:   V01
    CLEI:  IPUIAFNRAA

  NODE 0/1/5 : 8xGE SPA
    MAIN:  board type 044f
           68-2239-01 rev A0
           dev N/A
           S/N SAD0937022J
    PCA:   73-8557-03 rev A0
    PID:   SPA-8X1GE
    VID:   V01
    CLEI:  CNUIAH6AAA
```

```
NODE 0/RP0/CPU0 : RP
  MAIN:  board type 100002
         800-22921-10 rev B0
         dev 080366, 080181
         S/N SAD093507J8
  PCA:   73-8564-10 rev B0
  PID:   CRS-8-RP
  VID:   V01
  CLEI:  IPUCABWBAA
  ECI:   129507
  Board State : IOS XR RUN
  PLD:   Motherboard: 0x0038, Processor: 0x0038, Power: 0x0000
  MONLIB: QNXFFS Monlib Version 3.0
  ROMMON: Version 1.40(20050525:193559)  [CRS-1 ROMMON]

NODE 0/RP1/CPU0 : RP
  MAIN:  board type 100002
         800-22921-10 rev B0
         dev 080366, 080181
         S/N SAD093507JP
  PCA:   73-8564-10 rev B0
  PID:   CRS-8-RP
  VID:   V01
  CLEI:  IPUCABWBAA
  ECI:   129507
  Board State : IOS XR RUN
  PLD:   Motherboard: 0x0038, Processor: 0x0038, Power: 0x0000
  MONLIB: QNXFFS Monlib Version 3.0
  ROMMON: Version 1.40(20050525:193559)  [CRS-1 ROMMON]

NODE 0/SM0/SP : FC/S
  MAIN:  board type 400035
         800-23168-05 rev B0
         dev N/A
         S/N SAD0933081S
  PCA:   73-8682-05 rev B0
  PID:   CRS-8-FC/S
  VID:   V01
  CLEI:  IPUCABXBAA
  ECI:   129510
  Board State : IOS XR RUN
  PLD:   Motherboard: 0x001e, Processor: 0x0000, Power: N/A
  MONLIB: QNXFFS Monlib Version 3.0
  ROMMON: Version 1.40(20050525:193402)  [CRS-1 ROMMON]

NODE 0/SM1/SP : FC/S
  MAIN:  board type 400035
         800-23168-05 rev B0
         dev N/A
         S/N SAD09300492
  PCA:   73-8682-05 rev B0
  PID:   CRS-8-FC/S
  VID:   V01
  CLEI:  IPUCABXBAA
  ECI:   129510
  Board State : IOS XR RUN
  PLD:   Motherboard: 0x001e, Processor: 0x0000, Power: N/A
  MONLIB: QNXFFS Monlib Version 3.0
  ROMMON: Version 1.40(20050525:193402)  [CRS-1 ROMMON]

NODE 0/SM2/SP : FC/S
  MAIN:  board type 400035
         800-23168-05 rev B0
         dev N/A
```

```
        S/N SAD09330830
  PCA:   73-8682-05 rev B0
  PID:   CRS-8-FC/S
  VID:   V01
  CLEI:  IPUCABXBAA
  ECI:   129510
  Board State : IOS XR RUN
  PLD:   Motherboard: 0x001e, Processor: 0x0000, Power: N/A
  MONLIB: QNXFFS Monlib Version 3.0
  ROMMON: Version 1.40(20050525:193402)  [CRS-1 ROMMON]

NODE 0/SM3/SP : FC/S
  MAIN:  board type 400035
         800-23168-05 rev B0
         dev N/A
         S/N SAD0933081W
  PCA:   73-8682-05 rev B0
  PID:   CRS-8-FC/S
  VID:   V01
  CLEI:  IPUCABXBAA
  ECI:   129510
  Board State : IOS XR RUN
  PLD:   Motherboard: 0x001e, Processor: 0x0000, Power: N/A
  MONLIB: QNXFFS Monlib Version 3.0
  ROMMON: Version 1.40(20050525:193402)  [CRS-1 ROMMON]

Rack 0:

Fan Tray 0 : Fan Tray Upper
  MAIN:  board type 900160
         800-23275-05 rev A0
         dev N/A
         S/N TBA09370056
  PCA:   0-0-00 rev 00
  PID:   CRS-8-LCC-FAN-TR
  VID:   V01
  CLEI:  IPPQAGWJAB
  ECI:   133434

Fan Tray 1 : Fan Tray Lower
  MAIN:  board type 900160
         800-23275-05 rev A0
         dev N/A
         S/N TBA09370055
  PCA:   0-0-00 rev 00
  PID:   CRS-8-LCC-FAN-TR
  VID:   V01
  CLEI:  IPPQAGWJAB
  ECI:   133434

Rack 0:

Power Supply A :
  MAIN:  board type b00181
         341-112-01 rev C0
         dev N/A
         S/N TD109320008
  PCA:   0-0-00 rev 00
  PID:   CRS-8-AC-RECT
  VID:   V01
  CLEI:  IPP1D0WAAA
  ECI:   129500

Power Supply B :
```

```
  MAIN:  board type b00181
         341-112-01 rev C0
         dev N/A
         S/N TD10931000X
  PCA:   0-0-00 rev 00
  PID:   CRS-8-AC-RECT
  VID:   V01
  CLEI:  IPP1D0WAAA
  ECI:   129500

RACK   0 :
  MAIN:  board type 0001e4
         800-23271-04 rev F0
         dev 076763
         S/N TBA09370035
  PCA:   73-8696-03 rev A0
  PID:   CRS-8-LCC
  VID:   V01
  CLEI:  IPMEZ10BRA
  ECI:   446387
  RACK NUM: 0
```

Note Line cards are called modular services cards (MSCs).

In the following example, the **show diag** command displays information for a single system component:

```
RP/0/RP0/CPU0:router(admin)# show diag chassis

  RACK   0 :
    MAIN:  board type 0001e4
           800-23271-04 rev F0
           dev 076763
           S/N TBA09370035
    PCA:   73-8696-03 rev A0
    PID:   CRS-8-LCC
    VID:   V01
    CLEI:  IPMEZ10BRA
    ECI:   446387
    RACK NUM: 0
```

Displaying the Chassis Serial Numbers

Each chassis serial number must be defined during the configuration of multishelf routers. To view the actual serial number for each chassis in the system, enter the command **show diag chassis** in administration EXEC mode.

- Chassis serial numbers are displayed in the “Main” category for each chassis.
- “Rack Num” field displays the rack number assigned to that serial number.

For example:

```
RP/0/RP0/CPU0:router# admin
  RP/0/RP0/CPU0:router(admin)# show diag chassis

  RACK   0 :
```

```
  MAIN:  board type 0001e0
         800-24872
         dev 075078
         S/N TBA00000001
  PCA:   73-7640-05 rev 20
  PID:   CRS-16-LCC
  VID:   V01
  CLEI:  IPM6700DRA
  ECI:   445022
  RACK NUM: 0

RACK   1 :
  MAIN:  board type 0001e0
         800-24872-01 rev 20
         dev 075078
         S/N TBA00000002
  PCA:   73-7640-05 rev 20
  PID:   CRS-16-LCC
  VID:   V01
  CLEI:  IPM6700DRA
  ECI:   445022
  RACK NUM: 1

--MORE--
```

Displaying the Configured Chassis Serial Numbers

Enter the command **show running-config | include dsc** in administration EXEC mode to display the serial number configured for each rack number.

This command is used to verify that the configuration is correct. The serial numbers displayed are those entered by an operator. If this number is wrong because of an entry error, the number is still displayed, but the DSC does not recognize the chassis.

Note This command can also be entered in administration configuration mode.

For example:

```
RP/0/RP0/CPU0:router# admin
  RP/0/RP0/CPU0:router(admin)# show running-config | include dsc

  Building configuration...
  dsc serial TBA00000003 rack F0
  dsc serial TBA00000001 rack 0
  dsc serial TBA00000002 rack 1
  RP/0/RP0/CPU0:router(admin)#
```

Displaying Software and Hardware Information

The **show version** command displays a variety of system information, including the hardware and software versions, router uptime, boot settings (including the configuration register), and active software.

The following is sample output from the **show version** command:

```
RP/0/RP0/CPU0:router# show version

  Cisco IOS XR Software, Version 3.4.0[2I]
  Copyright (c) 2006 by cisco Systems, Inc.

  ROM: System Bootstrap, Version 1.40(20050525:193559) [CRS-1 ROMMON],

  router uptime is 1 week, 1 day, 17 hours, 1 minute
  System image file is "disk0:hfr-os-mbi-3.4.0/mbihfr-rp.vm"

  cisco CRS-8/S (7457) processor with 4194304K bytes of memory.
  7457 processor at 1197Mhz, Revision 1.2

  16 Packet over SONET/SDH network interface(s)
  16 SONET/SDH Port controller(s)
  2 Ethernet/IEEE 802.3 interface(s)
  16 GigabitEthernet/IEEE 802.3 interface(s)
  2043k bytes of non-volatile configuration memory.
  38079M bytes of hard disk.
  1000592k bytes of ATA PCMCIA card at disk 0 (Sector size 512 bytes).
  1000640k bytes of ATA PCMCIA card at disk 1 (Sector size 512 bytes).

  Package active on node 0/1/SP:
  hfr-diags, V 3.4.0[2I], Cisco Systems, at disk0:hfr-diags-3.4.0
      Built on Mon Mar 13 12:58:02 UTC 2006
      By iox8.cisco.com in /auto/ioxws48/production/3.4.0.2I/hfr/workspace for c8

  hfr-admin, V 3.4.0[2I], Cisco Systems, at disk0:hfr-admin-3.4.0
      Built on Mon Mar 13 11:46:36 UTC 2006
      By iox8.cisco.com in /auto/ioxws48/production/3.4.0.2I/hfr/workspace for c8

  hfr-base, V 3.4.0[2I], Cisco Systems, at disk0:hfr-base-3.4.0
      Built on Mon Mar 13 11:43:22 UTC 2006
      By iox8.cisco.com in /auto/ioxws48/production/3.4.0.2I/hfr/workspace for c8

  hfr-os-mbi, V 3.4.0[2I], Cisco Systems, at disk0:hfr-os-mbi-3.4.0
      Built on Mon Mar 13 11:27:02 UTC 2006
      By iox8.cisco.com in /auto/ioxws48/production/3.4.0.2I/hfr/workspace for c8

   --More--
```

Displaying Router Power Consumption

With the introduction of PLIMs and MSCs that consume higher power than before, and given the modular power available on a configurable number of power modules, it is possible that a fully loaded chassis can consume more power than available to the system. For this reason it is important to monitor your router power consumption and pay attention to any warnings or alarms regarding power.

Your router monitors the power necessary to run all cards in the system, and if the power requirements exceed the available power, syslog messages or alarms are displayed. Syslog messages can be displayed following two possible events:

- A board is powered up and a shortage of available power is detected.
- Available power becomes lower than the power consumed by inserted cards, for example because a power module is removed.

The following considerations are used when calculating the power consumption:

- Powering on an MSC or DRP adds to the power requirements of the chassis.
- Inserting or removing power modules affects the calculation of available power.
- Line cards are allowed to power up, before their power consumption is calculated.
- The power consumption of a SIP or SPA is calculated as though it is fully populated.
- RP, Switch Fabric, Fan tray, Fan controller and Alarm module power consumption is always added to the total chassis power usage regardless of whether they are physically present or not.
- The power of one power module is reserved for redundancy against a module failure (redundancy threshold), and thus subtracted from the calculation of available power.

Note For systems with modular power supplies, the total power availability is the sum of all power modules in both shelves *minus* one. This one power module is reserved to guard against a single module failure.

Note In a 4-Slot line card chassis, the total power available is the sum of all the power modules present (maximum of four).

Alarms and Messages

The following alarms can be raised:

- A major alarm is raised when the power consumption exceeds the power budget, and the alpha display on the alarm module is set to “PWR CRITICAL.”
- A minor alarm is raised when the redundancy threshold is crossed, and the alpha display is set to “PWR LOW.”
- A critical alarm is raised when there is a zone failure, and the alpha display is set to “ZONEX PWR FAIL,” where “X” is the zone number.

Syslog messages are displayed when a power event is registered.

Table 13: Syslog Messages Displayed on Systems with Modular Power Supplies

Event	Message
Power budget is exceeded	Power allotted to cards in this rack has exceeded the available rack power budget. Please check the 'show power' command to resolve this situation.
Power budget is restored	Power budget is now sufficient for rack power.
Power consumption exceeds the capacity of both shelves minus the capacity of one power module	Rack power is now being allotted from all power modules. Power module redundancy is no longer available, a single power module failure might result in card power loss.

Event	Message
Power consumption drops below the capacity of both shelves minus the capacity of one power module	Power allotment in this rack is now normal. Power module redundancy restored.

Table 14: Syslog Messages Displayed on Systems with Fixed Power Supplies

Event	Message
Zone power budget is exceeded	Power allotted in zone X has exceeded the available zone power budget. Please check the 'show power' command to resolve this situation.
Zone power budget is restored	Power budget for zone X is now sufficient for zone power.
Zone failure	Zone X has lost power. Check that power modules Ax and Bx are providing power.
Zone restoration	Zone X is now receiving power.

show power command Sample Output

Use the **show power** commands to display the total power available and the total power being consumed.

The **show power allotted** command displays the power allotted to the cards in the chassis. This example is from a system using modular power supplies:

```
RP/0/RP0/CPU0:router(admin)# show power allotted location 0/0/*

Sun Nov 18 22:00:51.176 UTC
nodeid = 0x2a00000f
Node              Card Type         State             PID               Power Allotted
----------------- ----------------- ----------------- ----------------- -----------------
0/0/*             FP-140G           POWERED UP        CRS-MSC-FP140       450.0W
0/0/PL0           14-10GbE          POWERED UP        14X10GBE-WL-XF      150.0W
```

The **show power capacity** command displays the power supplied to a rack. This example is from a system using fixed power supplies:

```
RP/0/RP1/CPU0:router(admin)# show power capacity rack 0

Tue Nov 20 19:43:30.458 OST
------------------------------------------------------------------------------
Rack 0: Cisco CRS Fixed AC Power System
------------------------------------------------------------------------------
Zone              Power Module      State             Zone Power Capacity
----------------- ----------------- ----------------- ---------------------
 Zone 1:           A[0]               NOT PRESENT        2500.0W
                   B[0]               OK
 Zone 2:           A[0]               NOT PRESENT        2500.0W
                   B[0]               OK
 Zone 3:           A[0]               NOT PRESENT        2500.0W
                   B[0]               OK
------------------------------------------------------------------------------
Total Rack Power Capacity:                                7500.0W
```

The **show power summary** displays a summary of the power consumption and availability for a rack. This example is from a system using modular power supplies:

```
RP/0/RP0/CPU0:router(admin)# show power summary rack 0

Sun Nov 18 22:02:40.434 UTC
Location          Power Capacity    Power Allotted    Power Available
----------------- ----------------- ----------------- ------------------
Rack :  0            7600.0W           1285.0W            6315.0W
```

Displaying SDR Node IDs and Status

In EXEC mode, the **show platform** command displays information for all nodes assigned to a secure domain router (SDR) . For each node, this information includes the host card type, the operational state, and the configuration state. To display information on a single node, enter the command with a node ID.

The syntax for the **show platform** command is:

show platform [*node-id*]

The following example displays the status for all nodes in the SDR to which you are connected:

```
RP/0/RP0/CPU0:router# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/0/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/2/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP0/CPU0      RP(Standby)     N/A             IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
```

Note Line cards are called modular services cards (MSCs).

The *node-id* appears in the *rack*/*slot*/*module* notation, and the *node-id* components are as follows:

- *rack* —In a single-shelf system the rack number is always “0.” In a multishelf system, the LCC rack number range is 0 to 255 and the FCC rack number range is F0 to F7.
- *slot* —Number of the physical slot in which the card is installed.
- *module* —Subslot number of a system hardware component.

Table 15: Node ID Components, on page 129 summarizes the *node-id* for each type of card.

Table 15: Node ID Components

Card Type (the card to which your are issuing commands)	Rack (always “0” in a single-shelf system)	Slot (the physical slot in which the card is installed)	Module (the entity on the card that is the target of the command)
Route processor	0–255	RP0 and RP1	CPU0

Card Type (the card to which your are issuing commands)	Rack (always "0" in a single-shelf system)	Slot (the physical slot in which the card is installed)	Module (the entity on the card that is the target of the command)
DRP	0-255	0–7 (8-slot chassis) 0–15 (16-slot chassis)	CPU0 or CPU1
MSC	0–255	0-3 (4-slot chassis) 0–7 (8-slot chassis) 0–15 (16-slot chassis)	Service processor (SP)
PLIM	0–255	0-3 (4-slot chassis) 0–7 (8-slot chassis) 0–15 (16-slot chassis)	CPU0
Cisco CRS-1 SPA Interface Processor (SIP)-800	0–255	0–7 (8-slot chassis) 0–15 (16-slot chassis)	CPU0
1-Port OC-192c/STM-64c Packet-over-SONET/SDH (POS) XFP SPA 4-Port OC-3c/STM-1 POS SPA 8-Port Gigabit Ethernet SPA	0–255	0–7 (8-slot chassis) 0–15 (16-slot chassis)	0–5 (SPA module number on the Cisco CRS-1 SIP-800)
Switch fabric module	0–255	SM0–SM3 (4-slot chassis) SM0–SM3 (8-slot chassis) SM0–SM7 (16-slot chassis)	SP
Alarm cards	0–255	AM0–AM1 (16-slot chassis)	SP
Fan controller cards	0–255	FC0–FC1 (16-slot chassis)	SP

Displaying Router Node IDs and Status

In administration EXEC mode, the **show platform** command displays information for all router nodes, which include nodes in all chassis and SDRs. In administration EXEC mode, the command display also includes additional node IDs such as those for fabric cards, alarm modules, and fan controllers. For each node, this information includes the host card type, the operational state, and the configuration state. To display information on a single node, enter the command with a node ID.

The syntax for the **show platform** command is:

show platform [*node-id*]

The following example displays the status for all nodes in a Cisco CRS-1 Multishelf System:

```
RP/0/RP0/CPU0:router(admin)# show platform

Node            Type            PLIM            State           Config State
--------------------------------------------------------------------------------
0/5/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
0/5/CPU0        MSC             4OC192-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
0/7/SP          DRP(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
0/7/CPU0        DRP(Active)     DRP-ACC         IOS XR RUN      PWR,NSHUT,MON
0/7/CPU1        DRP(Active)     DRP-ACC         IOS XR RUN      PWR,NSHUT,MON
0/14/SP         MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
0/14/CPU0       MSC             8-10GbE         IOS XR RUN      PWR,NSHUT,MON
0/RP0/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
0/RP1/CPU0      RP(Standby)     N/A             IOS XR RUN      PWR,NSHUT,MON
0/FC0/SP        LCC-FAN-CT(SP)  N/A             IOS XR RUN      PWR,NSHUT,MON
0/FC1/SP        LCC-FAN-CT(SP)  N/A             IOS XR RUN      PWR,NSHUT,MON
0/AM0/SP        ALARM(SP)       N/A             IOS XR RUN      PWR,NSHUT,MON
0/AM1/SP        ALARM(SP)       N/A             IOS XR RUN      PWR,NSHUT,MON
0/SM0/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
0/SM1/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
0/SM2/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
0/SM3/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
0/SM4/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
0/SM5/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
0/SM6/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
0/SM7/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
1/4/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
1/4/CPU0        MSC             4OC192-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
1/RP0/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
1/RP1/CPU0      RP(Standby)     N/A             IOS XR RUN      PWR,NSHUT,MON
1/FC0/SP        LCC-FAN-CT(SP)  N/A             IOS XR RUN      PWR,NSHUT,MON
1/FC1/SP        LCC-FAN-CT(SP)  N/A             IOS XR RUN      PWR,NSHUT,MON
1/AM0/SP        ALARM(SP)       N/A             IOS XR RUN      PWR,NSHUT,MON
1/SM0/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
1/SM1/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
1/SM3/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
1/SM4/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
1/SM5/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
1/SM6/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
1/SM7/SP        FC/M(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
F0/SM4/SP       FCC-SFC(SP)     FCC-FM-1S       IOS XR RUN      PWR,NSHUT,MON
F0/SM5/SP       FCC-SFC(SP)     FCC-FM-1S       IOS XR RUN      PWR,NSHUT,MON
F0/SM6/SP       FCC-SFC(SP)     FCC-FM-1S       IOS XR RUN      PWR,NSHUT,MON
F0/SM7/SP       FCC-SFC(SP)     FCC-FM-1S       IOS XR RUN      PWR,NSHUT,MON
F0/SC0/CPU0     FCC-SC(Active)  N/A             IOS XR RUN      PWR,NSHUT,MON
F0/SC1/CPU0     FCC-SC(Standby) N/A             IOS XR RUN      PWR,NSHUT,MON
F0/AM0/SP       ALARM(SP)       N/A             IOS XR RUN      PWR,NSHUT,MON
F0/AM1/SP       ALARM(SP)       N/A             IOS XR RUN      PWR,NSHUT,MON
F0/LM0/SP       FCC-LED(SP)     N/A             IOS XR RUN      PWR,NSHUT,MON
F0/LM1/SP       UNKNOWN(SP)     N/A             IN-RESET        PWR,NSHUT,MON
```

Note Line cards are called modular services cards (MSCs).

The *node-id* appears in the *rack*/*slot*/*module* notation, and the *node-id* components are as follows:

- *rack* —In a single-shelf system the rack number is always “0.” In a multishelf system, the LCC rack number range is 0 to 255 and the FCC rack number range is F0 to F7.
- *slot* —Number of the physical slot in which the card is installed.
- *module* —Subslot number of a system hardware component.

Table 15: Node ID Components, on page 129 summarizes the *node-id* argument for each type of card.

Displaying Router Environment Information

The **show environment** command displays hardware information for the system, including fan speeds, LED indications, power supply voltage and current information, and temperatures.

The syntax for the **show environment** command is:

show environment [*options*]

You can use the **show environment** command options to limit the detail in the command display. To view the command options, enter the **show environment ?** command. The following example shows the full environment status report:

```
RP/0/RP0/CPU0:router# show environment

  Temperature Information
  ----------------------------------------------

  R/S/I   Modules Sensor          Temp. (deg C)

  0/0/*     host  Inlet              23.0
            host  Hot                23.0
  0/3/*     host  Inlet              24.0
            host  Hot                33.0
  0/4/*     host  Inlet              24.5
            host  Hot                31.5
  0/5/*     host  Inlet              23.5
            host  Hot                30.5
  0/6/*     host  Hot                31.5
            host  Inlet              22.5
  0/7/*     host  Inlet              20.0
            host  Hot                29.5
  0/8/*     host  Inlet              20.5
            host  Hot                32.0

  Threshold Information
  ----------------------------------------------

  R/S/I   Modules Sensor          Minor           Major           Critical
                                  (Lo/Hi)         (Lo/Hi)         (Lo/Hi)

  0/0/*     host  InletTemp        --/  55         --/  60         --/  --
            host  HotTemp          --/  66         --/  69         --/  --
            host  PLIM_V4_1.6V     --/  --         --/  --         --/  --
            host  PLIM_V5_1.8V     --/  --         --/  --         --/  --
            host  PLIM_V3_2.5V     --/  --         --/  --         --/  --
            host  3.4V           2950/3500       2900/3600       --/  --
            host  5V             4800/5150       4700/5200       --/  --
            host  Mbus5V         4700/5300       4500/5500       --/  --
  0/3/*     host  InletTemp        --/  55         --/  60         --/  70
```

```
          host  HotTemp          --/  66          --/  69          --/  75
          host  PLIM_V3_1.5V     --/  --          --/  --          --/  --
          host  PLIM_V8_1.8V     --/  --          --/  --          --/  --
          host  PLIM_V7_2.5V     --/  --          --/  --          --/  --
          host  3.4V             --/  --          --/  --          --/  --
          host  5V             4800/5200        4700/5300        4600/5400
          host  Mbus5V         4700/5300        4600/5400        4500/5500
0/4/*     host  InletTemp        --/  55          --/  60          --/  70
          host  HotTemp          --/  66          --/  69          --/  75
          host  PLIM_V3_1.5V     --/  --          --/  --          --/  --
          host  PLIM_V8_1.8V     --/  --          --/  --          --/  --
          host  PLIM_V7_2.5V     --/  --          --/  --          --/  --
          host  PLIM_V6_1.5V     --/  --          --/  --          --/  --
          host  5V               --/  --          --/  --          --/  --
          host  3.4V             --/  --          --/  --          --/  --
          host  Mbus5V         4700/5300        4600/5400        4500/5500
0/5/*     host  InletTemp        --/  55          --/  60          --/  70
          host  HotTemp          --/  66          --/  69          --/  75
          host  PLIM_V3_1.5V     --/  --          --/  --          --/  --
          host  PLIM_V8_1.8V     --/  --          --/  --          --/  --
          host  PLIM_V7_2.5V     --/  --          --/  --          --/  --
          host  PLIM_V6_1.5V     --/  --          --/  --          --/  --
          host  5V               --/  --          --/  --          --/  --
          host  3.4V             --/  --          --/  --          --/  --
          host  Mbus5V         4700/5300        4600/5400        4500/5500
0/6/*     host  HotTemp          --/  66          --/  69          --/  75
          host  InletTemp        --/  55          --/  60          --/  70
          host  PLIM_V3_1.5V     --/  --          --/  --          --/  --
          host  PLIM_V8_1.8V     --/  --          --/  --          --/  --
          host  PLIM_V7_2.5V     --/  --          --/  --          --/  --
          host  3.4V             --/  --          --/  --          --/  --
          host  Mbus5V         4700/5300        4600/5400        4500/5500
0/7/*     host  InletTemp        --/  55          --/  60          --/  70
          host  HotTemp          --/  66          --/  69          --/  75
          host  PLIM_V3_1.5V     --/  --          --/  --          --/  --
          host  PLIM_V8_1.8V     --/  --          --/  --          --/  --
          host  PLIM_V7_2.5V     --/  --          --/  --          --/  --
          host  PLIM_V6_1.5V     --/  --          --/  --          --/  --
          host  5V               --/  --          --/  --          --/  --
          host  3.4V             --/  --          --/  --          --/  --
          host  Mbus5V         4700/5300        4600/5400        4500/5500
0/8/*     host  InletTemp        --/  55          --/  60          --/  70
          host  HotTemp          --/  66          --/  69          --/  75
          host  PLIM_V3_1.5V     --/  --          --/  --          --/  --
          host  PLIM_V8_1.8V     --/  --          --/  --          --/  --
          host  PLIM_V7_2.5V     --/  --          --/  --          --/  --
          host  3.4V             --/  --          --/  --          --/  --
          host  5V             4800/5200        4700/5300        4600/5400
          host  Mbus5V         4700/5300        4600/5400        4500/5500
Voltage Information
-------------------------------------------

R/S/I   Modules Sensor          Voltage (mV)    Margin

0/0/*     host  PLIM_V4_1.6V     1612           nominal
          host  PLIM_V5_1.8V     1804           nominal
          host  PLIM_V3_2.5V     2504           nominal
          host  3.4V             3296           nominal
          host  5V               5048           nominal
          host  Mbus5V           5048           n/a
0/3/*     host  PLIM_V3_1.5V     1496           nominal
          host  PLIM_V8_1.8V     1788           nominal
          host  PLIM_V7_2.5V     2492           nominal
          host  3.4V             3284           nominal
```

```
            host  5V               5000          nominal
            host  Mbus5V           5024          n/a
  0/4/*     host  PLIM_V3_1.5V     1500          nominal
            host  PLIM_V8_1.8V     1796          nominal
            host  PLIM_V7_2.5V     2488          nominal
            host  PLIM_V6_1.5V     1508          nominal
            host  5V               4976          nominal
            host  3.4V             3288          nominal
            host  Mbus5V           5048          n/a
  0/5/*     host  PLIM_V3_1.5V     1504          nominal
            host  PLIM_V8_1.8V     1792          nominal
            host  PLIM_V7_2.5V     2488          nominal
            host  PLIM_V6_1.5V     1504          nominal
            host  5V               4976          nominal
            host  3.4V             3284          nominal
            host  Mbus5V           4984          n/a
  0/6/*     host  PLIM_V3_1.5V     1496          nominal
            host  PLIM_V8_1.8V     1792          nominal
            host  PLIM_V7_2.5V     2476          nominal
            host  3.4V             3300          nominal
            host  Mbus5V           5016          n/a
  0/7/*     host  PLIM_V3_1.5V     1504          nominal
            host  PLIM_V8_1.8V     1796          nominal
            host  PLIM_V7_2.5V     2484          nominal
            host  PLIM_V6_1.5V     1504          nominal
            host  5V               4976          nominal
            host  3.4V             3276          nominal
            host  Mbus5V           4984          n/a
  0/8/*     host  PLIM_V3_1.5V     1496          nominal
            host  PLIM_V8_1.8V     1792          nominal
            host  PLIM_V7_2.5V     2492          nominal
            host  3.4V             3280          nominal
            host  5V               5000          nominal
            host  Mbus5V           5024          n/a
```

Displaying RP Redundancy Status

The **show redundancy** command displays the redundancy status of the route processors (RPs) . This command also displays the boot and switch-over history for the RPs.

The **show redundancy** operates in EXEC and administration EXEC mode.

In the following example, the **show redundancy** command displays the redundancy status for a redundant RP pair:

```
RP/0/RP0/CPU0:router# show redundancy

  This node (0/RP0/CPU0) is in ACTIVE role
  Partner node (0/RP1/CPU0) is in STANDBY role
  Standby node in 0/RP1/CPU0 is ready

  Reload and boot info
  ----------------------
  RP reloaded Fri Apr  9 03:44:28 2004: 16 hours, 51 minutes ago
  This node booted Fri Apr  9 06:19:05 2004: 14 hours, 16 minutes ago
  Last switch-over Fri Apr  9 06:53:18 2004: 13 hours, 42 minutes ago
  Standby node boot Fri Apr  9 06:54:25 2004: 13 hours, 41 minutes ago
  Standby node last not ready Fri Apr  9 20:35:23 2004: 0 minutes ago
  Standby node last ready Fri Apr  9 20:35:23 2004: 0 minutes ago
```

```
  There have been 2 switch-overs since reload
```

Displaying Field-Programmable Device Compatibility

The **show hw-module fpd** command displays field-programmable device (FPD) compatibility for all modules or a specific module.

The syntax for the **show hw-module fpd** command is:

show hw-module fpd location {**all** | *node-id*}

The **show hw-module fpd** operates in EXEC and administration EXEC mode.

The following example shows how to display FPD compatibility for all modules in the router:

```
RP/0/RSP0/CPU0:router# show hw-module fpd location all

==================================== ========================================
                                     Existing Field Programmable Devices
                                     ========================================
                                       HW                          Current SW Upg/
Location     Card Type               Version Type Subtype Inst   Version   Dng?
============ ======================= ======= ==== ======= ==== =========== ====
0/RSP0/CPU0  CRS1-SIP-800             1.0     lc   fpga3    0      1.23     Yes
                                                  fpga1    0      1.05      No
                                                  fpga2    0      3.08^     No
------------------------------------------------------------------------------
0/0/0       SPA-2XCHOC12/DS0           1.0     spa  rommon   0      2.02      No
                                               spa   fpga     0      1.36+     No
                                               spa   fpga2    0      1.00*     No
------------------------------------------------------------------------------
NOTES:
1.  One or more FPD needs an upgrade or a downgrade.  This can be accomplished
    using the "admin upgrade hw-module fpd" CLI.
2.  * One or more FPD is running minimum software version supported.
     It can be upgraded using the "admin> upgrade hw-module fpd <fpd> force location <loc>"
 CLI.
3.  + One or more FPD is running up-rev FPGA version. Downgrade is "OPTIONAL" in this case.

      It can be downgraded using the "admin> upgrade hw-module fpd <fpd> force location
<loc>" CLI.
4.  ^ One or more FPD will be intentionally skipped from upgrade using CLI with option "all"
 or during
     "Auto fpd". It can be upgraded only using the "admin> upgrade hw-module fpd <fpd>
location
     <loc>" CLI with exact location.
```

Note After Release 5.3.x, Upg/Dng? will display Yes only for upgrade.

The following example shows the FPD for which upgrage will be skipped.

```
RP/0/RP0/CPU0:router# show hw-module fpd location all

==================================== ========================================
                                     Existing Field Programmable Devices
```

```
                                      ========================================
                                        HW                        Current SW Upg/
Location     Card Type                Version Type Subtype Inst   Version   Dng?
============ ======================== ======= ==== ======= ==== =========== ====
0/SM1/SP     140G-4-S1S2S3              0.1  lc   rommonA   0    2.08        Yes
-------------------------------------------------------------------------------
                                             lc    rommon    0    2.08        Yes
-------------------------------------------------------------------------------
                                             lc    fpqa1     0    6.04^       No
-------------------------------------------------------------------------------
                                             lc    fpga2     0    4.01        No
-------------------------------------------------------------------------------
NOTES:
1. ^ One or more FPD will be intentionally skipped from upgrade using CLI with option "all"
 or during "Auto fpd".
    It can be upgraded only using the "admin> upgrade hw-module fpd <fpd> location <loc>"
CLI with exact location.


RP/0/RP0/CPU0:router# show hw-module fpd location 0/6/cpu0

Sun Apr 18 03:18:24.903 DST

==================================== ========================================
                                      Existing Field Programmable Devices
                                      ========================================
                                        HW                        Current SW Upg/
Location     Card Type                Version Type Subtype Inst   Version   Dng?
============ ======================== ======= ==== ======= ==== =========== ====
0/6/CPU0     CRS1-SIP-800               0.96  lc   fpga1   0       6.00     No
                                              lc   rommonA 0       2.100    No
                                              lc   rommon  0       2.100    No
-------------------------------------------------------------------------------
```

If the cards in the system do not meet the minimum requirements, the output contains a “NOTES” section that states how to upgrade the FPD image.

Table 16: show hw-module fpd Field Descriptions

Field	Description
Location	Location of the module in the *rack*/*slot*/*module* notation.
Card Type	Module part number.
HW Version	Hardware model version for the module.
Type	Hardware type. Can be one of the following types: • spa—Shared port adapter • lc—Line card

Field	Description
Subtype	FPD type. Can be one of the following types: • fabldr—Fabric downloader • fpga1—Field-programmable gate array • fpga2—Field-programmable gate array 2 • fpga3—Field-programmable gate array 3 • fpga4—Field-programmable gate array 4 • fpga5—Field-programmable gate array 5 • rommonA—Read-only memory monitor A • rommon—Read-only memory monitor B
Inst	FPD instance. The FPD instance uniquely identifies an FPD and is used by the FPD process to register an FPD.
Current SW Version	Currently running FPD image version.
Upg/Dng?	Specifies whether an FPD upgrade or downgrade is required. A downgrade is required in rare cases when the version of the FPD image has a higher major revision than the version of the FPD image in the current Cisco IOS XR software package.

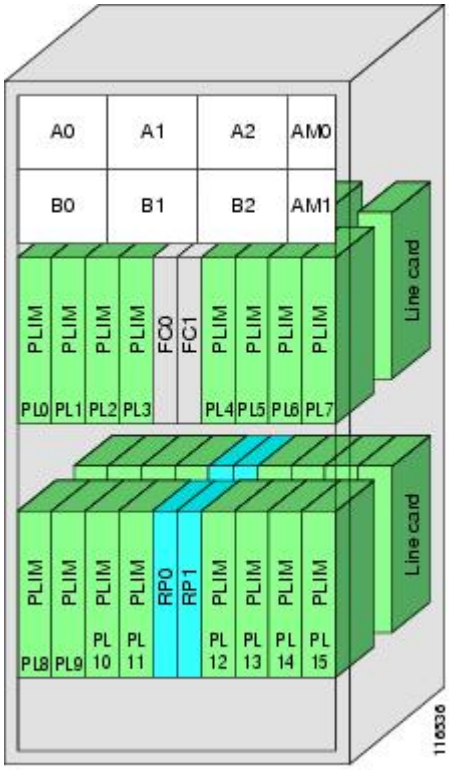
RP Redundancy and Switchover

This section describes RP redundancy and switchover commands and issues.

Establishing RP Redundancy

Your router has two slots for RPs: RP0 and RP1 (see Figure 4: Redundant Set of RPs Installed in Slots RP0 and RP1 in an 8-Slot Chassis, on page 138). These slots are configured for redundancy by default, and the redundancy cannot be eliminated. To establish RP redundancy, install RPs into both slots.

Figure 4: Redundant Set of RPs Installed in Slots RP0 and RP1 in an 8-Slot Chassis



Note: Illustration not to scale

Determining the Active RP in a Redundant Pair

During system startup, one RP in each redundant pair becomes the active RP. You can tell which RP is the active RP in the following ways:

- The active RP can be identified by the green Primary LED on the faceplate of the card. The active RP is indicated when the Primary LED is on. The alphanumeric LED display on the RP displays ACTV RP.
- The slot of the active RP is indicated in the CLI prompt. For example:

```
RP/0/RP1/CPU0:router#
```

  In this example, the prompt indicates that you are communicating with the active RP in slot RP1. See *Cisco IOS XR Getting Started Guide for the Cisco CRS Router* for a complete description of the CLI prompt.

- Enter the **show redundancy** command in EXEC mode to display a summary of the active and standby RP status. For example:

```
RP/0/RP0/CPU0:router# show redundancy

  This node (0/RP0/CPU0) is in ACTIVE role
  Partner node (0/RP1/CPU0) is in STANDBY role
  Standby node in 0/RP1/CPU0 is ready
```

```
Reload and boot info
----------------------
RP reloaded Fri Apr  9 03:44:28 2004: 16 hours, 51 minutes ago
This node booted Fri Apr  9 06:19:05 2004: 14 hours, 16 minutes ago
Last switch-over Fri Apr  9 06:53:18 2004: 13 hours, 42 minutes ago
Standby node boot Fri Apr  9 06:54:25 2004: 13 hours, 41 minutes ago
Standby node last not ready Fri Apr  9 20:35:23 2004: 0 minutes ago
Standby node last ready Fri Apr  9 20:35:23 2004: 0 minutes ago
There have been 2 switch-overs since reload
```

Role of the Standby RP

The second RP to boot in a redundant pair automatically becomes the standby RP. While the active RP manages the system and communicates with the user interface, the standby RP maintains a complete backup of the software and configurations for all cards in the system. If the active RP fails or goes off line for any reason, the standby RP immediately takes control of the system.

Summary of Redundancy Commands

RP redundancy is enabled by default in the Cisco IOS XR software, but you can use the commands described in Table 17: RP Redundancy Commands, on page 139 to display the redundancy status of the cards or force a manual switchover.

Table 17: RP Redundancy Commands

Command	Description
show redundancy	Displays the redundancy status of the RPs. This command also displays the boot and switch-over history for the RPs.
redundancy switchover	Forces a manual switchover to the standby RP. This command works only if the standby RP is installed and in the “ready” state.
show platform	Displays the status for node, including the redundancy status of the RP cards. In EXEC mode, this command displays status for the nodes assigned to the SDR. In administration EXEC mode, this command displays status for all nodes in the system.

Automatic Switchover

Automatic switchover from the active RP to the standby RP occurs only if the active RP encounters a serious system error, such as the loss of a mandatory process or a hardware failure. When an automatic switchover occurs, the RPs respond as follows:

- If a standby RP is installed and “ready” for switchover, the standby RP becomes the active RP. The original active RP attempts to reboot.
- If the standby RP is not in “ready” state, then both RPs reboot. The first RP to boot successfully assumes the role of active RP.

RP Redundancy During RP Reload

The **reload** command causes the active RP to reload the Cisco IOS XR software. When an RP reload occurs, the RPs respond as follows:

- If a standby RP is installed and "ready" for switchover, the standby RP becomes the active RP. The original active RP reboots and becomes the standby RP.
- If the standby RP is not in the "ready" state, then both RPs reboot. The first RP to boot successfully assumes the role of active RP.

Caution You should not use the **reload** command to force an RP switchover because the result could be a significant loss of router operations. Instead, use the **redundancy switchover** command to fail over to the standby RP, then use the **hw-module location** *node-id* **reload** command to reload the new standby RP.

Related Topics

Reloading, Shutting Down, or Power Cycling a Node, on page 143

Manual Switchover

You can force a manual switchover from the active RP to the standby RP using the **redundancy switchover** command.

If a standby RP is installed and ready for switchover, the standby RP becomes the active RP. The original active RP becomes the standby RP. In the following example, partial output for a successful redundancy switchover operation is shown:

```
RP/0/RP0/CPU0:router# show redundancy

  This node (0/RP0/CPU0) is in ACTIVE role
  Partner node (0/RP1/CPU0) is in STANDBY role
  Standby node in 0/RP1/CPU0 is ready

  RP/0/RP0/CPU0:router# redundancy switchover
  Updating Commit Database.  Please wait...[OK]
  Proceed with switchover 0/RP0/CPU0 -> 0/RP1/CPU0? [confirm]
  Initiating switch-over.
  RP/0/RP0/CPU0:router#

  <Your 'TELNET' connection has terminated>
```

In the preceding example, the Telnet connection is lost when the previously active RP resets. To continue management of the router, you must connect to the newly activated RP as shown in the following example:

```
User Access Verification

  Username: xxxxx
  Password: xxxxx
  Last switch-over Sat Apr 15 12:26:47 2009: 1 minute ago


RP/0/RP1/CPU0:router#
```

If the standby RP is not in “ready” state, the switchover operation is not allowed. In the following example, partial output for a failed redundancy switchover attempt is shown:

```
RP/0/RP0/CPU0:router# show redundancy

  Redundancy information for node 0/RP1/CPU0:
  ==========================================
  Node 0/RP0/CPU0 is in ACTIVE role
  Partner node (0/RP1/CPU0) is in UNKNOWN role

  Reload and boot info
  ----------------------
  RP reloaded Wed Mar 29 17:22:08 2009: 2 weeks, 2 days, 19 hours, 14 minutes ago
  Active node booted Sat Apr 15 12:27:58 2009: 8 minutes ago
  Last switch-over Sat Apr 15 12:35:42 2009: 1 minute ago
  There have been 4 switch-overs since reload

  RP/0/RP0/CPU0:router# redundancy switchover

  Switchover disallowed: Standby node is not ready.
```

Communicating with a Standby RP

The active RP automatically synchronizes all system software, settings, and configurations with the standby RP.

If you connect to the standby RP through the console port, you can view the status messages for the standby RP. The standby RP does not display a CLI prompt, so you cannot manage the standby card while it is in standby mode.

If you connect to the standby RP through the management Ethernet port, the prompt that appears is for the active RP, and you can manage the router the same as if you had connected through the management Ethernet port on the active RP.

CPAK

CPAKs are the Cisco's innovation for 100G pluggable optics, which is built with the industry leading smallest form factor, in full compliant with IEEE802.3ae specification for 100GE-SR10, -LR4, and can interoperate with all IEEE 802.3ba compliant CFP-SR10 or CFP-LR4 100G optics.

Modes Supported on CPAKs

This table clearly lists the modes supported with the relevant PID:

CPAK (PID)	Modes Supported
CPAK-100G-SR10	100 GE, 10 GE, 40 GE
CPAK-100G-LR	100 GE

CPAK (PID)	Modes Supported
CPAK-10X10G-LR	100 GE

Power saving mode

8x100GE Line card consists of 4 Slices (0,1,2,3). Each slice has two physical ports. Slice-1, 2 and 3 can be configured into power save mode. Power save option is not applicable to Slice-0. Use the **hw-module power saving** command to configure the required slice to power saving mode.

Once a slice is configured in the power saving mode, the interfaces will be deleted and hence all traffic passing through the interfaces will be dropped.

Table 18: Slice-Port mapping table

Slice 1	Ports 2,3
Slice 2	Ports 4,5
Slice 3	Ports 6,7

To configure the power save option

This task enables the user to configure the power save option.

SUMMARY STEPS

1. **admin**
2. **configure**
3. **hw-module power saving location** *location* **slice** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**configure**	
Step 3	**hw-module power saving location** *location* **slice** *number* **Example:** `RP/0/RP0/CPU0:router (admin-config) #` **`hw-module power saving location 0/1/CPU0 slice 3`**	Configures the power saving option for the specified slice. The available options are Slice1, 2, 3. **Note** Power save option is not applicable for Slice 0.

What to do next

Use the **show plat slices** command to get the status of the slices.

Reloading, Shutting Down, or Power Cycling a Node

Use the commands described in this section to reload the Cisco IOS XR software on the active RP or on any specified node in the system. This section also describes the commands used to administratively shut down a node and power a node on or off.

Table 19: Commands to Reload, Shut Down, or Power Cycle a Node, on page 143 summarizes the commands described in this section.

Table 19: Commands to Reload, Shut Down, or Power Cycle a Node

Command	Description
hw-module location *node-id* **power disable**	This command administratively turns the power off for a node. It is entered in administration configuration mode. The changes do not take effect until you enter the **commit** command. To power on a node, use the **no** form of this command. **Note** This command cannot be used to disable power on the RP from which the command is entered.
hw-module location *node-id* **reload**	This command works in EXEC mode and reloads the Cisco IOS XR software on a specific node or all nodes. To specify all nodes, enter the **all** keyword in place of the *node-id* argument. The node reloads with the current running configuration and active software set for that node.

Reloading the Active RP

The **reload** command causes the active RP to reload the Cisco IOS XR software according to the configuration register setting. This setting determines how the active RP acts when reloaded.

This section contains instructions to reload the Cisco IOS XR software and return to EXEC mode. For instructions to use the **reload** command for entering ROM Monitor bootstrap mode, see *ROM Monitor Configuration Guide for Cisco CRS Routers*.

Caution Because the **reload** command causes the active RP to go off line and either reload the Cisco IOS XR software or enter ROM Monitor mode, the router experiences a loss of service unless a redundant standby RP is installed and in "ready" state. To display the status of the standby RP, use the **show redundancy** command in EXEC mode.

SUMMARY STEPS

1. **show redundancy**
2. **admin**
3. **show variables boot**
4. (Optional) **config-register** *register-value*
5. **admin**

6. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show redundancy** **Example:** `RP/0/RP0/CPU0:router# show redundancy`	Displays the RP redundancy status. • If a standby RP is in "ready" redundancy state, the **reload** command also causes the router to gracefully fail over to the standby RP.
Step 2	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 3	**show variables boot** **Example:** `RP/0/RP0/CPU0:router(admin)# show variables boot`	Displays the configuration register setting. • Enter this command in administration EXEC mode. • For normal operations, the configuration register setting is 0x102 or 0x2102, which causes the active RP to reload the Cisco IOS XR software. • Verify that the configuration register setting is 0x102 or 0x2102. If it is not, complete Step 4, on page 144 to reset the configuration register to 0x102 or 0x2102. **Note** For instructions on how to enter ROM Monitor bootstrap mode, see *ROM Monitor Configuration Guide for Cisco CRS Routers*.
Step 4	(Optional) **config-register** *register-value* **Example:** `RP/0/RP0/CPU0:router(admin)# config-register 0x102`	Sets the configuration register to the respective value. This step is necessary only if the register is not set to the respective value (0x102 or 0x2102) in the running configuration. You can use either 0x102 or 0x2102. Both these values specify the same functionality, as bit 13 in 0x2102 is not significant for Cisco IOS XR software.
Step 5	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 6	**reload** **Example:** `RP/0/RP0/CPU0:router# reload`	Reloads the active RP according to the configuration register setting. • If the setting is 0x102 or 0x2102 , then the RP reloads the Cisco IOS XR software. • If the standby RP is in "ready" redundancy state, the router switches over to the standby RP. • If a standby RP is not installed or not in a "ready" state, the router experiences a loss of service while the active RP is reloading the Cisco IOS XR software.

Flash Disk Recovery

When an RP or DRP is power cycled or experiences an ungraceful reset, the boot disk (PCMCIA flash disk used to boot the card) may experience a file-system corruption. If this occurs, an error message is displayed and the RP or DRP fails to boot. The corrupted flash disk is automatically reformatted and the Cisco IOS XR software is restored from the designated system controller (DSC) for the system.

For example, if a flash disk for an RP or DRP is corrupted, the RP or DRP fails to boot and the following error message is displayed:

```
############################################################
              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 XR Software for the Cisco XR Cisco CRS Router-mbirp,
Copyright (c) 2009 by Cisco Systems, Inc.
Unable to mount /disk0:, filesystem is corrupted.
Check fsck log at /tmp/chkfs_fd0.log
init: special_commands:wait for disk0: failed
```

If this occurs, then the flash disk is automatically reformatted and the Cisco IOS XR software is restored to the flash disk.

Note If the flash disk is badly damaged and cannot be reformatted, the disk must be replaced.

If the corrupted flash disk is the DSC, then the router fails over to the standby DSC. If no standby DSC is installed, then the system fails to boot.

Using Controller Commands to Manage Hardware Components

The **controller** , **controllers** , and **show controllers** commands are used to manage and display settings for various hardware components, including the switch fabric management, Ethernet control plane, and interface manager. These commands are primarily diagnostic and related to driver-level details. The information available with these commands varies widely and is hardware specific.

For information on the use of these commands, see *Interface and Hardware Component Command Reference for Cisco CRS Routers*.

Formatting Hard Drives, Flash Drives, and Other Storage Devices

To format a storage device on the router, use the **format** command in EXEC mode.

Caution Formatting a storage device deletes all data on that device.

The following command syntax is used:

format *filesystem:* [*options*]

Table 20: format command Syntax Description, on page 146 describes the **format** command syntax.

Table 20: format command Syntax Description

Variable	Description
filesystem	Specifies the memory device to format. The supported file systems are: • **bootflash:** • **compactflash:** • **flash:** • **harddisk:** • **harddiska:** • **disk0:** • **disk1:** Enter **format ?** to see the devices supported on your router.
options	Enter **format** *filesystem:* **?** to see the available options. For more information, see *System Management Command Reference for isco CRS Routers*.

In the following example, the **format** command is used to format the hard disk:

```
RP/0/RP0/CPU0:router# format harddisk:
```

Removing and Replacing Cards

This section describes card replacement issues and procedures.

Removing Line Cards, MSCs, or PLIMs

Line cards, modular services cards (MSCs), and physical layer interface modules (PLIMs) are designed for online insertion and removal (OIR). The service processing functions are provided on the MSC, and the physical line interface is provided on a separate card that connects the physical lines to the MSC.

The OIR feature allows you to remove and replace cards without removing power to the card or chassis. Removing a card interrupts all traffic passing through the card, but it does not remove the card configuration.

When you remove a card, the configuration remains for all interfaces, but the interfaces do not appear in the output of the **show interfaces** command. You can view interface configurations by entering the **show running-config** command. The following example shows how the configuration appears when a card is removed:

```
RP/0/RP0/CPU0:router# show running-config

  Building configuration...
  hostname router
  router ospf 3269
   area 0
    interface POS0/3/0/0
     cost 20
  !
  interface preconfigure POS0/3/0/0
   ipv4 address 10.10.50.1 255.255.255.0
  !
  interface preconfigure POS0/3/0/1
   description POS0/3/0/1
   shutdown
  !
  interface preconfigure POS0/3/0/2
   description POS0/3/0/2
   shutdown
  !
  interface preconfigure POS0/3/0/3
   description POS0/3/0/3
   shutdown
  !
```

In this example, the MSC in slot 3 is removed, and the interface configuration for all four interfaces changes to “interface preconfigure.” However, the “router ospf” reference to a slot 3 interface does not change. If you replace a PLIM with another PLIM that uses the same media type and port count, the configuration becomes active on the replacement card.

To remove the configuration for a slot after a card is removed, use the **no interface preconfigure** command to remove all interface configuration statements for that card in the running configuration. In addition, search the configuration for any references to the removed interfaces, such as the “router ospf” reference to slot 3 in the preceding example.

To remove the configuration for a slot when a card is installed, use the **no interface** command to remove all interface configuration statements for that card in the running configuration. In addition, search the configuration for any references to the removed interfaces.

Each PLIM supports a specific media type (Packet over SONET/SDH [POS] or Ethernet, for example) and port count. If you replace a PLIM with one that supports a different media type or port count, you should review the configuration and revise it to support the replacement PLIM .

Replacing an MSC

When you replace an MSC, the guidelines in the Removing Line Cards, MSCs, or PLIMs , on page 146 apply. Because only one type of MSC exists, no special procedures are required for card removal and replacement.

Replacing a Line Card or PLIM with the Same Media Type and Port Count

When you replace a line card or PLIM with a card that is of the same media type and has the same port count as the replaced card, the guidelines in the Removing Line Cards, MSCs, or PLIMs , on page 146 apply. Because the replacement card is of the same media type and port count, no special procedures are required for card removal and replacement.

Replacing a Line Card or PLIM with the Same Media Type and a Different Port Count

When you replace a line card or PLIM with a card that is of the same media type with a different port count, the guidelines in Removing Line Cards, MSCs, or PLIMs , on page 146 apply.

If the new card has a greater port count than the replaced card, the configuration applies to the corresponding lower port numbers, and the ports that did not exist on the replaced card have no configuration and come up in the shutdown state.

If the new card supports fewer ports, the existing configuration for the corresponding number of ports on the new card set is applied. The previous configuration for the removed ports remains in interface preconfigure state, as shown in the following example:

```
RP/0/RP0/CPU0:router# show running-config

  Building configuration...
  hostname rtp-gsr1
  interface POS0/3/0/0
   ipv4 address 10.10.50.1 255.255.255.0
  !
  interface preconfigure POS0/3/0/1
   description POS0/3/0/1
   shutdown
  !
  interface preconfigure POS0/3/0/2
   description POS0/3/0/2
   shutdown
  !
  interface preconfigure POS0/3/0/3
   description POS0/3/0/3
   shutdown
  !
```

In the preceding example, a four-port card has been replaced with a single-port card. The configuration from port 1 on the four-port card is applied to the single port on the replacement card, and the remaining port configurations change to “interface preconfigure.” To remove the configuration for the missing interfaces, use the **no interface preconfigure** command. In addition, search for and remove any configuration references to the removed interfaces.

Whenever you replace a line card or PLIM with the same media type and a different port count, review the running configuration in the router and revise the configuration as necessary.

Replacing a Line Card or PLIM with a Different Media Type

When you replace a line card or PLIM with a card that is of a different media type (for example, if you replace a POS PLIM with an Ethernet PLIM), the guidelines in Removing Line Cards, MSCs, or PLIMs , on page 146 apply. Review the running configuration in the router and revise the configuration as necessary for the new media type.

Real Time Power Monitoring

Real Time Power monitoring feature consolidates the power consumption values into a common interface. The user can now know the real time power being consumed on the individual slots and the router as a whole.

Advantages

With real time power monitoring, power consumption is maintained at slot level granularity. The user can identify to which power consuming slab the system belongs to, and can take business decisions accordingly.

Card support

Real Time Power is supported on the following cards:

- Cisco CRS Modular Services card 400G
- Cisco CRS Modular Services card 200G
- Cisco CRS Series 16 Slots Fabric Card / Multi (400G)
- Cisco CRS Series 16 Slots Fabric Card / Multi (200G)
- Cisco CRS Series 16 Slots Fabric Card / Single (400G)
- Cisco CRS Series 8 Slots Fabric Card / Single (400G)
- Cisco CRS Series 40x10GE Interface Module
- Cisco CRS Series 4x100GE Interface Module
- Cisco CRS 2X100GE (CPAK) and 5X40GE (QSFP+) LAN/OTN Flexible Interface Module

Examples: Breakout and Power saving options

The following are the examples for the **power save** and **breakout** options:

Power saving mode

Configuring the power saving option:

```
admin
 config
  hw-module power saving location 0/0/CPU0 slice 3
 !

show platform slices
Line Card       Slice   Config          Status
0/0/CPU0        0       Power on        Completed
                1       Power on        Completed
                2       Power on        Completed
                3       Power saving    Completed
```

Breakout option

Configuring the breakout option:

```
config
 hw-module location 0/0/CPU0 port 0 breakout 10xTenGigE
!
```

show command output indicating the breakout ports:

```
RP/0/RSP0/CPU0:TD02#show ipv4 interface brief | include Hun
```

```
Sun Sep  7 15:59:33.446 PST
HundredGigE0/0/0/0             34.34.34.2      Down                Down
HundredGigE0/0/0/1             100.0.1.1       Up                  Up
HundredGigE0/0/0/2             unassigned      Up                  Up
HundredGigE0/0/0/3             unassigned      Up                  Up
HundredGigE0/0/0/4             unassigned      Shutdown            Down
HundredGigE0/0/0/5             unassigned      Shutdown            Down
HundredGigE0/0/0/6             unassigned      Shutdown            Down
HundredGigE0/0/0/7             unassigned      Shutdown            Down


RP/0/RSP0/CPU0:router(config)#hw-module location 0/0/CPU0 port 2 breakout 10xTenGigE
RP/0/RSP0/CPU0:router(config)#commit

RP/0/RSP0/CPU0:router#show ipv4 interface brief | include Ten
TenGigE0/0/0/2/0               unassigned      Shutdown            Down
TenGigE0/0/0/2/1               unassigned      Shutdown            Down
TenGigE0/0/0/2/2               unassigned      Shutdown            Down
TenGigE0/0/0/2/3               unassigned      Shutdown            Down
TenGigE0/0/0/2/4               unassigned      Shutdown            Down
TenGigE0/0/0/2/5               unassigned      Shutdown            Down
TenGigE0/0/0/2/6               unassigned      Shutdown            Down
TenGigE0/0/0/2/7               unassigned      Shutdown            Down
TenGigE0/0/0/2/8               unassigned      Shutdown            Down
TenGigE0/0/0/2/9               unassigned      Shutdown            Down
```

Removing and Replacing Cisco 16-Slot Line Card Chassis Switch Fabric Cards

16-slot LCCs support two switch fabric cards: the CRS-16-FC/S and the CRS-16-FC/M. The CRS-16-FC/S switch fabric card provides the Stage 1, 2, and 3 switch fabric for one fabric plane in a standalone Cisco CRS-1 Carrier Routing System 16-Slot Line Card Chassis. The CRS-16-FC/M switch fabric card provides the Stage 1 and 3 switch fabric for one fabric plane in a Cisco CRS-1 LCC within a multishelf system.

The Cisco CRS-1 16-Slot LCC can support the maximum throughput with seven of the eight fabric planes. To prevent traffic loss, we recommend that you shut the power down on a fabric plane for a switch fabric card before you remove it. If a switch fabric card is removed with the power on, the card is not harmed, but some traffic may be lost. When the replacement card is inserted, you can restore the power to the fabric plane and bring up the replacement card. This section describes how to properly remove and replace Cisco CRS-16-FC/S and Cisco CRS-16-FC/M cards for upgrades or repairs.

Note The process of removing and replacing cards while the router power is on is called *online insertion and removal (OIR)*. This procedure removes power to a specific slot before the switch fabric card is replaced. The power remains on for all other slots.

Tip For more information about switch fabric cards, see *Related Topics*.

Note This procedure does not apply when starting the router for the first time or after a power cycle or reload.

Before you begin

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- You must log in as root-system before starting the procedure. To confirm your login status, use the **show user group** command:

```
RP/0/RP0/CPU0:router# show user group

  root-system, cisco-support
```

To confirm your login status including root, use the **show user all** | **include root** command:

```
RP/0/RP0/CPU0:router# show user all | include root

  Groups: root-system, cisco-support
  Task:              root-lr  : READ    WRITE    EXECUTE    DEBUG (reserved)
  Task:          root-system  : READ    WRITE    EXECUTE    DEBUG (reserved)
```

SUMMARY STEPS

1. **admin**
2. **show platform**
3. **show controllers fabric plane all**
4. **admin**
5. **controllers fabric plane** *plane_number* **shutdown**
6. **commit**
7. **end**
8. **show controllers fabric plane all**
9. **admin**
10. **hw-module power disable location** *node-id*
11. **show platform**
12. When the fabric card state changes to UNPOWERED, replace the fabric card.
13. **admin**
14. **no hw-module power disable location** *node-id*
15. **show platform**
16. **admin**
17. **no controllers fabric plane** *plane_number* **shutdown**
18. **show controllers fabric plane all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:**	Enters administration EXEC mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router# admin`	
Step 2	**show platform** **Example:** `RP/0/RP0/CPU0:router(admin)# show platform`	Displays all cards on the router. • Allows you to identify a fabric card (identified with an SM prefix). • The number following the SM prefix identifies the corresponding fabric plane, as follows: • Slot SM0: fabric plane 0 • Slot SM1: fabric plane 1 • Slot SM2: fabric plane 2 • Slot SM3: fabric plane 3 • Slot SM4: fabric plane 4 • Slot SM5: fabric plane 5 • Slot SM6: fabric plane 6 • Slot SM7: fabric plane 7
Step 3	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the status of each fabric plane.
Step 4	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

Step 5	**controllers fabric plane** *plane_number* **shutdown**
	Example:
	`RP/0/RP0/CPU0:router(admin-config)# controllers fabric`
	`plane 0 shutdown`
		Shuts down the fabric plane.

Step 6	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.
Step 7	**end** **Example:**	Exits administration configuration mode and returns to administration EXEC mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(admin-config)# end`	
Step 8	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the status of each fabric plane. • The *Admin State* and *Oper State* columns should read DOWN.
Step 9	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 10	**hw-module power disable location** *node-id* **Example:** `RP/0/RP0/CPU0:router(admin-config)# hw-module power disable location 0/SM0/SP`	Sets the target configuration to remove power from the fabric card.
Step 11	**show platform** **Example:** `RP/0/RP0/CPU0:router(admin)# show platform`	Displays the status of all cards on the router. • Check the *State* column for the status of the fabric card. • Do not continue to the next step until the status in the *State* column changes to UNPOWERED. • It takes some time for the card to shut down. Repeat the **show platform** command to check the card state.
Step 12	When the fabric card state changes to UNPOWERED, replace the fabric card.	Replaces the physical card.
Step 13	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 14	**no hw-module power disable location** *node-id* **Example:** `RP/0/RP0/CPU0:router(admin-config)# no hw-module power disable location 0/SM0/SP`	Sets the target configuration to restore power to the fabric card.
Step 15	**show platform** **Example:** `RP/0/RP0/CPU0:router(admin)# show platform`	Displays the status of all cards on the router. • Check the *State* column for the status of the fabric card. • Do not continue to the next step until the status in the *State* column changes to IOS XR RUN.

	Command or Action	Purpose
		• It takes some time for the card to start up. Repeat the **show platform** command to check the card state.
Step 16	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 17	**no controllers fabric plane** *plane_number* **shutdown** **Example:** `RP/0/RP0/CPU0:router(admin-config)# no controllers fabric plane 0 shutdown`	Sets the target configuration to bring up the fabric plane.
Step 18	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the fabric plane status. The *Admin State* and *Oper State* columns should read UP.

Related Topics

Additional References, on page 175

Examples

The following example shows the commands and command responses for replacing a a 16-slot LCC fabric card:

```
RP/0/RP1/CPU0:router# admin
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM4/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM5/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM6/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM7/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

    Flags: P - plane admin down,       p - plane oper down

           C - card admin down,        c - card  oper down
           L - link port admin down,   l - linkport oper down
           A - asic admin down,        a - asic oper down
           B - bundle port admin Down, b - bundle port oper down
```

```
          I - bundle admin down,      i - bundle oper down
          N - node admin down,        n - node down
          o - other end of link down  d - data down
          f - failed component downstream
          m - plane multicast down

   Plane  Admin   Oper

   Id     State   State
   --------------------
   0      UP      UP
   1      UP      UP
   2      UP      UP
   3      UP      UP
   4      UP      UP
   5      UP      UP
   6      UP      UP
   7      UP      UP

RP/0/RP1/CPU0:router(admin)# configure
RP/0/RP1/CPU0:router(admin-config)# controllers fabric plane 0 shutdown
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:15:09.265 : fsdb_aserver[173]: %FABRIC-FSDB-1-PLANE_UPDO
  WN : Plane 0 state changed to DOWN:
  RP/0/RP1/CPU0:Oct  5 02:15:09.319 : config[65734]: %MGBL-LIBTARCFG-6-ADMIN_COMMI
  T : Administration configuration committed by user 'jim'.

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

    Flags: P - plane admin down,       p - plane oper down
           C - card admin down,        c - card  oper down
           L - link port admin down,   l - linkport oper down
           A - asic admin down,        a - asic oper down
           B - bundle port admin Down, b - bundle port oper down
           I - bundle admin down,      i - bundle oper down
           N - node admin down,        n - node down
           o - other end of link down  d - data down
           f - failed component downstream
           m - plane multicast down

   Plane  Admin   Oper
   Id     State   State
   --------------------
   0      DOWN    DOWN
   1      UP      UP
   2      UP      UP
   3      UP      UP
   4      UP      UP
   5      UP      UP
   6      UP      UP
   7      UP      UP

RP/0/RP1/CPU0:router(admin)# configure
RP/0/RP1/CPU0:router(admin-config)# hw-module power disable location 0/SM0/SP
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:18:24.774 : config[65734]: %MGBL-LIBTARCFG-6-COMMIT : Co
  nfiguration committed by user 'jim'.   Use 'show configuration commit changes 10
  00000142' to view the changes.
  RP/0/RP1/CPU0:router(config)#LC/0/1/CPU0:Oct  5 02:18:26.873 : fabricq_mgr[
  127]: %FABRIC-FABRICQ-3-FI_UNCORR_ERROR : fabricq: Major error in Fabric Interfa
  ce : RS Uncorrectable errors on Fabricq ASIC 0 link 3
```

```
  RP/0/RP1/CPU0:Oct  5 02:18:28.959 : shelfmgr[284]: %PLATFORM-SHELFMGR-3-POWERDOW
  N_RESET : Node 0/SM0/SP is powered off due to admin power off request

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             UNPOWERED       NPWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM4/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM5/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM6/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM7/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
```

When the state of the fabric card changes to UNPOWERED, replace the fabric card.

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(admin-config)# no hw-module power disable location 0/SM0/SP
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:19:30.472 : config[65734]: %MGBL-LIBTARCFG-6-COMMIT : Co
  nfiguration committed by user 'jim'.   Use 'show configuration commit changes 10
  00000143' to view the changes.
  RP/0/RP1/CPU0:router(config)#RP/0/RP1/CPU0:Oct  5 02:19:42.747 : shelfmgr[2
  84]: %PLATFORM-MBIMGR-7-IMAGE_VALIDATED : 0/SM0/SP: MBI tftp:/hfr-os-mbi-3.4.0/
  sp/mbihfr-sp.vm validated

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             MBI-BOOTING     PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM4/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM5/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM6/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM7/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             MBI-RUNNING     PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM4/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM5/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
```

```
  0/SM6/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM7/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM4/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM5/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM6/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM7/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

RP/0/RP1/CPU0:router(admin)# configure

  SP/0/SM0/SP:Oct  5 02:20:19.102 : init[65541]: %OS-INIT-7-MBI_STARTED : total time
  7.678 seconds
  SP/0/SM0/SP:Oct  5 02:20:21.361 : insthelper[60]: %INSTALL-INSTHELPER-7-PKG_DOWN
  LOAD : MBI running; starting software download
  SP/0/SM0/SP:Oct  5 02:22:23.458 : init[65541]: %OS-INIT-7-INSTALL_READY : total
  time 132.060 seconds
  SP/0/SM0/SP:Oct  5 02:22:39.329 : sfe_drvr[108][120]: Board revision : 0x06.
  SP/0/SM0/SP:Oct  5 02:22:47.306 : sfe_drvr[108]: %FABRIC-FABRIC_DRVR-6-ASIC_IN
  ITIALIZED : Fabric ASICs initialized
  SP/0/SM0/SP:Oct  5 02:23:06.316 : alphadisplay[100]: %PLATFORM-ALPHA_DISPLAY-6-CHANGE :
  Alpha display on node 0/SM0/SP changed to IOS-XR   in state default

RP/0/RP1/CPU0:router(admin-config)# no controllers fabric plane 0 shutdown
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:25:15.736 : fsdb_aserver[173]: %FABRIC-FSDB-1-PLANE_UPDO
  WN : Plane 0 state changed to UP:
  RP/0/RP1/CPU0:Oct  5 02:25:15.759 : config[65734]: %MGBL-LIBTARCFG-6-ADMIN_COMMI
  T : Administration configuration committed by user 'jim'.

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

    Flags: P - plane admin down,       p - plane oper down
           C - card admin down,        c - card  oper down

           L - link port admin down,   l - linkport oper down
           A - asic admin down,        a - asic oper down
           B - bundle port admin Down, b - bundle port oper down
           I - bundle admin down,      i - bundle oper down
           N - node admin down,        n - node down
           o - other end of link down  d - data down
           f - failed component downstream
           m - plane multicast down

   Plane  Admin   Oper
   Id     State   State
   --------------------
   0      UP      UP
   1      UP      UP
   2      UP      UP
   3      UP      UP
   4      UP      UP
```

```
5       UP       UP
6       UP       UP
7       UP       UP
```

Related Topics

Additional References, on page 175

Removing and Replacing 8-Slot Line Card Chassis Switch Fabric Cards

Each CRS-8-FC/S switch fabric card provides the Stage 1, 2, and 3 switch fabric for two fabric planes in a Cisco CRS-1 8-Slot Line Card Chassis.

The 8-Slot LCC can support the maximum throughput with seven of the eight fabric planes. However, because each CRS-8-FC/S switch fabric card hosts two fabric planes, replacing a fabric card does reduce the maximum throughput and impacts router traffic if the router is operating at maximum capacity. To minimize traffic loss, we recommend that you shut the power down for the switch fabric card before you remove it. If a switch fabric card is removed with power on, the card is not harmed, but the traffic impact may be greater than if the card power were removed. When the replacement card is inserted, you can restore the power and bring up the replacement card. This section describes how to properly remove and replace a Cisco CRS-8-FC/S switch fabric card for upgrades or repairs.

The process of removing and replacing cards while the router power is on is called *online insertion and removal (OIR)*. This procedure removes power to a specific slot before the switch fabric card is replaced. The power remains on for all other slots.

For more information about switch fabric cards, see *Related Topics*.

Note This procedure does not apply when starting the router for the first time or after a power cycle or reload.

Before you begin

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- You must log in as root-system before starting the procedure. To confirm your login status, use the **show user group** command:

```
RP/0/RP0/CPU0:router# show user group

  root-system, cisco-support
```

To confirm your login status including root, use the **show user all | include root** command:

```
RP/0/RP0/CPU0:router# show user all | include root

  Groups: root-system, cisco-support
  Task:              root-lr  : READ    WRITE    EXECUTE    DEBUG (reserved)
  Task:          root-system  : READ    WRITE    EXECUTE    DEBUG (reserved)
```

SUMMARY STEPS

1. **admin**
2. **show platform**
3. **show controllers fabric plane all**
4. **admin**
5. **controllers fabric plane** *plane_number* **shutdown**
6. **controllers fabric plane** *plane_number* **shutdown**
7. **commit**
8. **end**
9. **show controllers fabric plane all**
10. **admin**
11. **hw-module power disable location** *node-id*
12. **commit**
13. **end**
14. **show platform**
15. When the fabric card state changes to UNPOWERED, replace the fabric card.
16. **admin**
17. **no hw-module power disable location** *node-id*
18. **commit**
19. **end**
20. **show platform**
21. **admin**
22. **no controllers fabric plane** *plane_number* **shutdown**
23. **no controllers fabric plane** *plane_number* **shutdown**
24. **commit**
25. **end**
26. **show controllers fabric plane all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

Step 2	**show platform**
	Example:
	`RP/0/RP0/CPU0:router(admin)# show platform`
		Displays all cards on the router.
		• Allows you to identify a fabric card (identified with an SM prefix).
		• The number following the SM prefix identifies the corresponding fabric planes, as follows:
		  • Slot SM0: fabric planes 0 and 1
		  • Slot SM1: fabric planes 2 and 3
		  • Slot SM2: fabric planes 4 and 5

	Command or Action	Purpose
		• Slot SM3: fabric planes 6 and 7
Step 3	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the status of each fabric plane.
Step 4	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

Step 5	**controllers fabric plane** *plane_number* **shutdown**
	Example:
	`RP/0/RP0/CPU0:router(admin-config)# controllers fabric plane 0`
	`shutdown`
		Shuts down one of the two fabric planes on a CRS-8-FC/S card.
		• Before removing a CRS-8-FC/S card, shut down both planes for the fabric card. The fabric planes are assigned to fabric cards as follows:
		• Slot SM0: fabric planes 0 and 1
		• Slot SM1: fabric planes 2 and 3
		• Slot SM2: fabric planes 4 and 5
		• Slot SM3: fabric planes 6 and 7

Step 6	**controllers fabric plane** *plane_number* **shutdown**
	Example:
	`RP/0/RP0/CPU0:router(admin-config)# controllers fabric plane 1`
	`shutdown`
		Shuts down one of the two fabric planes on a CRS-8-FC/S card.
		• Shut down the companion plane to the plane shut down in the previous step.

Step 7	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.
Step 8	**end** **Example:** `RP/0/RP0/CPU0:router(admin-config)# end`	Exits administration configuration mode and returns to administration EXEC mode.
Step 9	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the status of each fabric plane. The *Admin State* and *Oper State* columns should read DOWN for both of the shutdown planes.

	Command or Action	Purpose
Step 10	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 11	**hw-module power disable location** *node-id* **Example:** `RP/0/RP0/CPU0:router(admin-config)# hw-module power disable location 0/SM0/SP`	Sets the target configuration to remove power from the fabric card.
Step 12	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.
Step 13	**end** **Example:** `RP/0/RP0/CPU0:router(admin-config)# end`	Exits administration configuration mode and returns to administration EXEC mode.
Step 14	**show platform** **Example:** `RP/0/RP0/CPU0:router(admin)# show platform`	Displays the status of all cards on the router. • Check the *State* column for the status of the fabric card. • Do not continue to the next step until the status in the *State* column changes to UNPOWERED. • It takes some time for the card to shut down. Repeat the **show platform** command to check the card state.
Step 15	When the fabric card state changes to UNPOWERED, replace the fabric card.	Replaces the physical card.
Step 16	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 17	**no hw-module power disable location** *node-id* **Example:** `RP/0/RP0/CPU0:router(admin-config)# no hw-module power disable location 0/SM0/SP`	Sets the target configuration to restore power to the fabric card.
Step 18	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.

	Command or Action	Purpose
Step 19	**end** **Example:** `RP/0/RP0/CPU0:router(admin-config)# end`	Exits administration configuration mode and returns to administration EXEC mode.
Step 20	**show platform** **Example:** `RP/0/RP0/CPU0:router(admin)# show platform`	Displays the status of all cards on the router. • Check the *State* column for the status of the fabric card. • Do not continue to the next step until the status in the *State* column changes to IOS XR RUN. • It takes some time for the card to start up. Repeat the **show platform** command to check the card state.
Step 21	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

Step 22	**no controllers fabric plane** *plane_number* **shutdown**
	Example:
	`RP/0/RP0/CPU0:router(admin-config)# no controllers fabric`
	`plane 0 shut`
		Sets the target configuration to bring up one of the two fabric planes on the card.

Step 23	**no controllers fabric plane** *plane_number* **shutdown**
	Example:
	`RP/0/RP0/CPU0:router(admin-config)# no controllers fabric`
	`plane 1 shut`
		Sets the target configuration to bring up one of the two fabric planes on the card.

Step 24	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.
Step 25	**end** **Example:** `RP/0/RP0/CPU0:router(admin-config)# end`	Exits administration configuration mode and returns to administration EXEC mode.
Step 26	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the fabric plane status. The *Admin State* and *Oper State* columns should read UP for both fabric planes on the fabric card.

Related Topics

Additional References, on page 175

Examples

The following example shows the commands and command responses for replacing an 8-slot LCC fabric card:

```
RP/0/RP1/CPU0:router# admin
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON


RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

  Flags: P - plane admin down,       p - plane oper down

         C - card admin down,        c - card  oper down
         L - link port admin down,   l - linkport oper down
         A - asic admin down,        a - asic oper down
         B - bundle port admin Down, b - bundle port oper down
         I - bundle admin down,      i - bundle oper down
         N - node admin down,        n - node down
         o - other end of link down  d - data down
         f - failed component downstream
         m - plane multicast down

   Plane  Admin   Oper

   Id     State   State
   --------------------
   0      UP      UP
   1      UP      UP
   2      UP      UP
   3      UP      UP
   4      UP      UP
   5      UP      UP
   6      UP      UP
   7      UP      UP

RP/0/RP1/CPU0:router(admin)# configure
RP/0/RP1/CPU0:router(admin-config)# controllers fabric plane 0 shutdown
RP/0/RP1/CPU0:router(admin-config)# controllers fabric plane 1 shutdown
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:15:09.265 : fsdb_aserver[173]: %FABRIC-FSDB-1-PLANE_UPDO
  WN : Plane 0 state changed to DOWN:
  RP/0/RP1/CPU0:Oct  5 02:15:09.265 : fsdb_aserver[173]: %FABRIC-FSDB-1-PLANE_UPDO
  WN : Plane 1 state changed to DOWN:
  RP/0/RP1/CPU0:Oct  5 02:15:09.319 : config[65734]: %MGBL-LIBTARCFG-6-ADMIN_COMMI
  T : Administration configuration committed by user 'jim'.

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

  Flags: P - plane admin down,       p - plane oper down
         C - card admin down,        c - card  oper down
```

```
        L - link port admin down,   l - linkport oper down
        A - asic admin down,        a - asic oper down
        B - bundle port admin Down, b - bundle port oper down
        I - bundle admin down,      i - bundle oper down
        N - node admin down,        n - node down
        o - other end of link down  d - data down
        f - failed component downstream
        m - plane multicast down

   Plane  Admin   Oper
   Id     State   State
   --------------------
   0      DOWN    DOWN
   1      DOWN    DOWN
   2      UP      UP
   3      UP      UP
   4      UP      UP
   5      UP      UP
   6      UP      UP
   7      UP      UP

RP/0/RP1/CPU0:router(admin)# configure
RP/0/RP1/CPU0:router(admin-config)# hw-module power disable location 0/SM0/SP
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:18:24.774 : config[65734]: %MGBL-LIBTARCFG-6-COMMIT : Co
  nfiguration committed by user 'jim'.   Use 'show configuration commit changes 10
  00000142' to view the changes.
  RP/0/RP1/CPU0:router(config)#LC/0/1/CPU0:Oct  5 02:18:26.873 : fabricq_mgr[
  127]: %FABRIC-FABRICQ-3-FI_UNCORR_ERROR : fabricq: Major error in Fabric Interfa
  ce : RS Uncorrectable errors on Fabricq ASIC 0 link 3
  RP/0/RP1/CPU0:Oct  5 02:18:28.959 : shelfmgr[284]: %PLATFORM-SHELFMGR-3-POWERDOW
  N_RESET : Node 0/SM0/SP is powered off due to admin power off request

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             UNPOWERED       NPWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
```

When the state for the fabric card changes to UNPOWERED, replace the fabric card.

```
RP/0/RP1/CPU0:router(admin)# configure
RP/0/RP1/CPU0:router(admin-config)# no hw-module power disable location 0/SM0/SP
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:19:30.472 : config[65734]: %MGBL-LIBTARCFG-6-COMMIT : Co
  nfiguration committed by user 'jim'.   Use 'show configuration commit changes 10
  00000143' to view the changes.
  RP/0/RP1/CPU0:router(config)#RP/0/RP1/CPU0:Oct  5 02:19:42.747 : shelfmgr[2
  84]: %PLATFORM-MBIMGR-7-IMAGE_VALIDATED : 0/SM0/SP: MBI tftp:/hfr-os-mbi-3.4.0/
  sp/mbihfr-sp.vm validated

RP/0/RP1/CPU0:router(admin-config)# end
```

```
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             MBI-BOOTING     PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             MBI-RUNNING     PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON


RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

RP/0/RP1/CPU0:router(admin)# configure

  RP/0/RP1/CPU0:router(admin-config)#SP/0/SM0/SP:Oct  5 02:20:19.102 : init[6
  5541]: %OS-INIT-7-MBI_STARTED : total time 7.678 seconds
  SP/0/SM0/SP:Oct  5 02:20:21.361 : insthelper[60]: %INSTALL-INSTHELPER-7-PKG_DOWN
  LOAD : MBI running; starting software download
  SP/0/SM0/SP:Oct  5 02:22:23.458 : init[65541]: %OS-INIT-7-INSTALL_READY : total
  time 132.060 seconds
  SP/0/SM0/SP:Oct  5 02:22:39.329 : sfe_drvr[108][120]: Board revision : 0x06.
  SP/0/SM0/SP:Oct  5 02:22:47.306 : sfe_drvr[108]: %FABRIC-FABRIC_DRVR-6-ASIC_IN
  ITIALIZED : Fabric ASICs initialized
  SP/0/SM0/SP:Oct  5 02:23:06.316 : alphadisplay[100]: %PLATFORM-ALPHA_DISPLAY-6-CHANGE :
  Alpha display on node 0/SM0/SP changed to IOS-XR   in state default

RP/0/RP1/CPU0:router(admin-config)# no controllers fabric plane 0 shutdown
RP/0/RP1/CPU0:router(admin-config)# no controllers fabric plane 1 shutdown
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:25:15.736 : fsdb_aserver[173]: %FABRIC-FSDB-1-PLANE_UPDO
  WN : Plane 0 state changed to UP:
  RP/0/RP1/CPU0:Oct  5 02:25:15.736 : fsdb_aserver[173]: %FABRIC-FSDB-1-PLANE_UPDO
  WN : Plane 1 state changed to UP:
  RP/0/RP1/CPU0:Oct  5 02:25:15.759 : config[65734]: %MGBL-LIBTARCFG-6-ADMIN_COMMI
  T : Administration configuration committed by user 'jim'.

RP/0/RP1/CPU0:router(admin-config)# end
```

```
  RP/0/RP1/CPU0:Oct  5 02:25:41.891 : config[65734]: %MGBL-SYS-5-CONFIG_I : Config
  ured from console by jim

RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

  Flags: P - plane admin down,       p - plane oper down
         C - card admin down,        c - card  oper down

         L - link port admin down,   l - linkport oper down
         A - asic admin down,        a - asic oper down
         B - bundle port admin Down, b - bundle port oper down
         I - bundle admin down,      i - bundle oper down
         N - node admin down,        n - node down
         o - other end of link down  d - data down
         f - failed component downstream
         m - plane multicast down

   Plane  Admin   Oper
   Id     State   State
   --------------------
   0      UP      UP
   1      UP      UP
   2      UP      UP
   3      UP      UP
   4      UP      UP
   5      UP      UP
   6      UP      UP
   7      UP      UP
```

Related Topics

Additional References, on page 175

Removing and Replacing Cisco 4-Slot Line Card Chassis Switch Fabric Cards

The Cisco CRS-4-FC switch fabric card provides the Stage 1, 2, and 3 switch fabric for one fabric plane in a Cisco CRS-1 4-Slot Line Card Chassis.

The Cisco CRS-1 4-Slot LCC can support the maximum throughput with three of the four fabric planes. To prevent traffic loss, we recommend that you shut the power down on a fabric plane for a switch fabric card before you remove it. If a switch fabric card is removed with the power on, the card is not harmed, but some traffic may be lost. When the replacement card is inserted, you can restore the power to the fabric plane and bring up the replacement card. This section describes how to properly remove and replace Cisco CRS-4-FC switch fabric cards for upgrades or repairs.

Note At least two planes of the switch fabric (an even plane and an odd plane) must be active at all times for the Cisco CRS-1 4-slot line card chassis to operate. Otherwise, the switch fabric fails, causing a system failure.

The process of removing and replacing cards while the router power is on is called *online insertion and removal (OIR)*. This procedure removes power to a specific slot before the switch fabric card is replaced. The power remains on for all other slots.

For more information about switch fabric cards, see the hardware documentation listed in the Related Documents, on page 175.

Note This procedure does not apply when starting the router for the first time or after a power cycle or reload.

Before you begin

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- You must log in as root-system before starting the procedure. To confirm your login status, use the **show user group** command:

```
RP/0/RP0/CPU0:router# show user group

  root-system, cisco-support
```

To confirm your login status including root, use the **show user all | include root** command:

```
RP/0/RP0/CPU0:router# show user all | include root

  Groups: root-system, cisco-support
  Task:              root-lr  : READ    WRITE    EXECUTE    DEBUG (reserved)
  Task:          root-system  : READ    WRITE    EXECUTE    DEBUG (reserved)
```

SUMMARY STEPS

1. **admin**
2. **show platform**
3. **show controllers fabric plane all**
4. **configure**
5. **controllers fabric plane** *plane_number* **shutdown**
6. **commit**
7. **end**
8. **show controllers fabric plane all**
9. **configure**
10. **hw-module power disable location** *node-id*
11. **commit**
12. **end**
13. **show platform**
14. When the fabric card state changes to UNPOWERED, replace the fabric card.
15. **configure**
16. **no hw-module power disable location** *node-id*
17. **commit**
18. **end**
19. **show platform**
20. **configure**
21. **no controllers fabric plane** *plane_number* **shutdown**

22. **commit**
23. **end**
24. **show controllers fabric plane all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

Step 2	**show platform**
	Example:
	`RP/0/RP0/CPU0:router(admin)# show platform`
		Displays all cards on the router.
		• Allows you to identify a fabric card (identified with an SM prefix).
		• The number following the SM prefix identifies the corresponding fabric plane, as follows:
		  • Slot SM0: fabric plane 0
		  • Slot SM1: fabric plane 1
		  • Slot SM2: fabric plane 2
		  • Slot SM3: fabric plane 3

Step 3	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the status of each fabric plane.
Step 4	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.

Step 5	**controllers fabric plane** *plane_number* **shutdown**
	Example:
	`RP/0/RP0/CPU0:router(admin-config)# controllers fabric`
	`plane 0 shutdown`
		Shuts down the fabric plane.

Step 6	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.
Step 7	**end** **Example:** `RP/0/RP0/CPU0:router(admin-config)# end`	Exits administration configuration mode and returns to administration EXEC mode.

	Command or Action	Purpose
Step 8	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the status of each fabric plane. • The *Admin State* and *Oper State* columns should read DOWN.
Step 9	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 10	**hw-module power disable location** *node-id* **Example:** `RP/0/RP0/CPU0:router(admin-config)# hw-module power disable location 0/SM0/SP`	Sets the target configuration to remove power from the fabric card.
Step 11	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.
Step 12	**end** **Example:** `RP/0/RP0/CPU0:router(admin-config)# end`	Exits administration configuration mode and returns to administration EXEC mode.
Step 13	**show platform** **Example:** `RP/0/RP0/CPU0:router(admin)# show platform`	Displays the status of all cards on the router. • Check the *State* column for the status of the fabric card. • Do not continue to the next step until the status in the *State* column changes to UNPOWERED. • It takes some time for the card to shut down. Repeat the **show platform** command to check the card state.
Step 14	When the fabric card state changes to UNPOWERED, replace the fabric card.	Replaces the physical card.
Step 15	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.
Step 16	**no hw-module power disable location** *node-id* **Example:** `RP/0/RP0/CPU0:router(admin-config)# no hw-module power disable location 0/SM0/SP`	Sets the target configuration to restore power to the fabric card.

	Command or Action	Purpose
Step 17	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.
Step 18	**end** **Example:** `RP/0/RP0/CPU0:router(admin-config)# end`	Exits administration configuration mode and returns to administration EXEC mode.
Step 19	**show platform** **Example:** `RP/0/RP0/CPU0:router(admin)# show platform`	Displays the status of all cards on the router. • Check the *State* column for the status of the fabric card. • Do not continue to the next step until the status in the *State* column changes to IOS XR RUN. • It takes some time for the card to start up. Repeat the **show platform** command to check the card state.
Step 20	**configure** **Example:** `RP/0/RP0/CPU0:router(admin)# configure`	Enters administration configuration mode.

Step 21	**no controllers fabric plane** *plane_number* **shutdown**
	Example:
	`RP/0/RP0/CPU0:router(admin-config)# no controllers fabric`
	`plane 0 shutdown`
		Sets the target configuration to bring up the fabric plane.

Step 22	**commit** **Example:** `RP/0/RP0/CPU0:router(admin-config)# commit`	Commits the target configuration to the router running configuration.
Step 23	**end** **Example:** `RP/0/RP0/CPU0:router(admin-config)# end`	Exits administration configuration mode and returns to administration EXEC mode.
Step 24	**show controllers fabric plane all** **Example:** `RP/0/RP0/CPU0:router(admin)# show controllers fabric plane all`	Displays the fabric plane status. The *Admin State* and *Oper State* columns should read UP.

Examples

The following example shows the commands and command responses for replacing a 4-slot LCC switch fabric card:

```
RP/0/RP1/CPU0:router# admin
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

    Flags: P - plane admin down,       p - plane oper down

           C - card admin down,        c - card  oper down
           L - link port admin down,   l - linkport oper down
           A - asic admin down,        a - asic oper down
           B - bundle port admin Down, b - bundle port oper down
           I - bundle admin down,      i - bundle oper down
           N - node admin down,        n - node down
           o - other end of link down  d - data down
           f - failed component downstream
           m - plane multicast down

   Plane  Admin   Oper

   Id     State   State
   --------------------
   0      UP      UP
   1      UP      UP
   2      UP      UP
   3      UP      UP

RP/0/RP1/CPU0:router(admin)# configure
RP/0/RP1/CPU0:router(admin-config)# controllers fabric plane 0 shutdown
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:15:09.265 : fsdb_aserver[173]: %FABRIC-FSDB-1-PLANE_UPDO
  WN : Plane 0 state changed to DOWN:
  RP/0/RP1/CPU0:Oct  5 02:15:09.319 : config[65734]: %MGBL-LIBTARCFG-6-ADMIN_COMMI
  T : Administration configuration committed by user 'jim'.

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

    Flags: P - plane admin down,       p - plane oper down
           C - card admin down,        c - card  oper down
           L - link port admin down,   l - linkport oper down
           A - asic admin down,        a - asic oper down
           B - bundle port admin Down, b - bundle port oper down
           I - bundle admin down,      i - bundle oper down
           N - node admin down,        n - node down
           o - other end of link down  d - data down
           f - failed component downstream
```

```
          m - plane multicast down

   Plane  Admin   Oper
   Id     State   State
   -------------------
   0      DOWN    DOWN
   1      UP      UP
   2      UP      UP
   3      UP      UP

RP/0/RP1/CPU0:router(admin)# configure
RP/0/RP1/CPU0:router(admin-config)# hw-module power disable location 0/SM0/SP
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:18:24.774 : config[65734]: %MGBL-LIBTARCFG-6-COMMIT : Co
  nfiguration committed by user 'jim'.   Use 'show configuration commit changes 10
  00000142' to view the changes.
  RP/0/RP1/CPU0:router(config)#LC/0/1/CPU0:Oct  5 02:18:26.873 : fabricq_mgr[
  127]: %FABRIC-FABRICQ-3-FI_UNCORR_ERROR : fabricq: Major error in Fabric Interfa
  ce : RS Uncorrectable errors on Fabricq ASIC 0 link 3
  RP/0/RP1/CPU0:Oct  5 02:18:28.959 : shelfmgr[284]: %PLATFORM-SHELFMGR-3-POWERDOW
  N_RESET : Node 0/SM0/SP is powered off due to admin power off request

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             UNPOWERED       NPWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
```

When the state of the fabric card changes to UNPOWERED, replace the fabric card.

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(admin-config)# no hw-module power disable location 0/SM0/SP
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:19:30.472 : config[65734]: %MGBL-LIBTARCFG-6-COMMIT : Co
  nfiguration committed by user 'jim'.   Use 'show configuration commit changes 10
  00000143' to view the changes.
  RP/0/RP1/CPU0:router(config)#RP/0/RP1/CPU0:Oct  5 02:19:42.747 : shelfmgr[2
  84]: %PLATFORM-MBIMGR-7-IMAGE_VALIDATED : 0/SM0/SP: MBI tftp:/hfr-os-mbi-3.4.0/
  sp/mbihfr-sp.vm validated

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show platform

  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             MBI-BOOTING     PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
```

```
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

  RP/0/RP1/CPU0:router(admin)# show platform


 Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             MBI-RUNNING     PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON

RP/0/RP1/CPU0:router(admin)# show platform


  Node            Type            PLIM            State           Config State
  ------------------------------------------------------------------------------
  0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
  0/1/CPU0        MSC             16OC48-POS/DPT  IOS XR RUN      PWR,NSHUT,MON
  0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM0/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM1/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM2/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON
  0/SM3/SP        FC/S(SP)        N/A             IOS XR RUN      PWR,NSHUT,MON


RP/0/RP1/CPU0:router(admin)# configure

  SP/0/SM0/SP:Oct  5 02:20:19.102 : init[65541]: %OS-INIT-7-MBI_STARTED : total time
  7.678 seconds
  SP/0/SM0/SP:Oct  5 02:20:21.361 : insthelper[60]: %INSTALL-INSTHELPER-7-PKG_DOWN
  LOAD : MBI running; starting software download
  SP/0/SM0/SP:Oct  5 02:22:23.458 : init[65541]: %OS-INIT-7-INSTALL_READY : total
  time 132.060 seconds
  SP/0/SM0/SP:Oct  5 02:22:39.329 : sfe_drvr[108][120]: Board revision : 0x06.
  SP/0/SM0/SP:Oct  5 02:22:47.306 : sfe_drvr[108]: %FABRIC-FABRIC_DRVR-6-ASIC_IN
  ITIALIZED : Fabric ASICs initialized
  SP/0/SM0/SP:Oct  5 02:23:06.316 : alphadisplay[100]: %PLATFORM-ALPHA_DISPLAY-6-CHANGE :
  Alpha display on node 0/SM0/SP changed to IOS-XR   in state default

RP/0/RP1/CPU0:router(admin-config)# no controllers fabric plane 0 shutdown
RP/0/RP1/CPU0:router(admin-config)# commit

  RP/0/RP1/CPU0:Oct  5 02:25:15.736 : fsdb_aserver[173]: %FABRIC-FSDB-1-PLANE_UPDO
  WN : Plane 0 state changed to UP:
  RP/0/RP1/CPU0:Oct  5 02:25:15.759 : config[65734]: %MGBL-LIBTARCFG-6-ADMIN_COMMI
  T : Administration configuration committed by user 'jim'.

RP/0/RP1/CPU0:router(admin-config)# end
RP/0/RP1/CPU0:router(admin)# show controllers fabric plane all

    Flags: P - plane admin down,       p - plane oper down
           C - card admin down,        c - card  oper down

           L - link port admin down,   l - linkport oper down
           A - asic admin down,        a - asic oper down
           B - bundle port admin Down, b - bundle port oper down
           I - bundle admin down,      i - bundle oper down
           N - node admin down,        n - node down
           o - other end of link down  d - data down
```

```
        f - failed component downstream
        m - plane multicast down

Plane  Admin   Oper
Id     State   State
--------------------
0      UP      UP
1      UP      UP
2      UP      UP
3      UP      UP
```

Upgrading the CPU Controller Bits

Use this procedure to upgrade the CPU controller bits on all nodes that are installed in the router or on a specific node.

SUMMARY STEPS

1. **admin**
2. **upgrade cpuctrlbits** {**all** | **location** *node-id*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 2	**upgrade cpuctrlbits** {**all** | **location** *node-id*} **Example:** `RP/0/RP0/CPU0:router(admin)# upgrade cpuctrlbits all`	Upgrades the CPU controller bits on all nodes in the router. Use the **location** *node-id* keyword and argument to upgrade the CPU controller bits on a specific node.

Examples

The following example shows how to upgrade the CPU controller bits on all nodes in a router:

```
RP/0/RP0/CPU0:router# admin
RP/0/RP0/CPU0:router(admin)# upgrade cpucrtlbits all

Please do not power cycle, reload the router or reset any nodes until all upgrades are
completed.
Please check the syslog to make sure that all nodes are upgraded successfully.
If you need to perform multiple upgrades, please wait for current upgrade to be completed
before proceeding to another upgrade. Failure to do so may render the cards under upgrade
to be unusable.
```

Additional References

The following sections provide references related to hardware management on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Cisco IOS XR hardware commands	Hardware Redundancy and Node Administration Commands on *the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*
Cisco IOS XR hardware documentation	See Cisco Carrier Routing System Install and Upgrade Guides at: http://www.cisco.com/en/US/products/ps5763/prod_installation_guides_list.html
Information about getting started with Cisco IOS XR software	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*
ROM Monitor	*ROM Monitor Configuration Guide for Cisco CRS Routers*
Cisco IOS XR command master list	*Cisco IOS XR Commands Master List for the Cisco CRS Router*
Information about user groups and task IDs	*Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

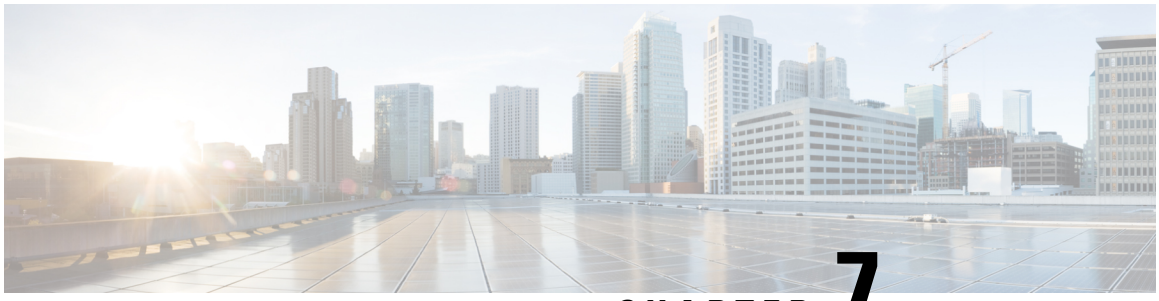
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 7

Configuring Flexible Command Line Interface

This module describes how to configure and use flexible command line interface (CLI) configuration groups.

Table 21: Feature History for Configuring Flexible CLI Configuration Groups

Release	Modification
Release 4.3.1	Flexible CLI configuration groups were introduced.

This module contains these topics:

Information About Flexible CLI Configuration Groups

Flexible command line interface (CLI) configuration groups provide the ability to minimize repetitive configurations by defining a series of configuration statements in a configuration group, and then applying this group to multiple hierarchical levels in the router configuration tree.

Flexible CLI configuration groups utilize regular expressions that are checked for a match at multiple submodes of the configuration tree based on where the group is applied within the hierarchy. If a match is found at a configuration submode, the corresponding configuration defined in the group is inherited within the matched submode.

Flexible CLI configuration groups also provide an auto-inheritance feature. Auto-inheritance means that any change done to a CLI configuration group is automatically applied to the configuration in any matched submodes that have an apply-group at that hierarchical level. This allows you to make a configuration change or addition once, and have it applied automatically in multiple locations, depending on where you have applied the flexible CLI configuration group.

Flexible Configuration Restrictions

Note these restrictions while using flexible configuration groups:

- Flexible CLI configuration groups are not supported in administration configurations and corresponding apply-groups are not supported in administration configurations.
- Use of preconfigured interfaces in configuration groups is not supported.
- Downgrading from an image that supports configuration groups to an image that does not support them is not supported.
- Access lists, quality of service and route policy configurations do not support the use of configuration groups. Configurations such as these are not valid:

```
group g-not-supported
 ipv4 access-list ...
 !
 ipv6 access-list ...
 !
 ethernet-service access-list ...
 !
 class-map ...
 !
 policy-map ...
 !
 route-policy ...
 !
end-group
```

You can, however, reference such configurations, as shown in this example:

```
group g-reference-ok
 router bgp 6500
  neighbor 7::7
   remote-as 65000
   bfd fast-detect
   update-source Loopback300
   graceful-restart disable
   address-family ipv6 unicast
    route-policy test1 in
    route-policy test2 out
    soft-reconfiguration inbound always
   !
  !
 !
 interface Bundle-Ether1005
   bandwidth 10000000
   mtu 9188
   service-policy output input_1
   load-interval 30
 !
end-group
```

- Some regular expressions are not supported within groups. For example, '?', '|' and '$,' are not supported within groups. Also some characters such as /d and /w are not supported.

- The choice operator "|" to express multiple match expressions within a regular expression is not supported. For example, these expressions are not supported:

  `Gig.*|Gig.*\..*`—To match on either Gigabit Ethernet main interfaces or Gigabit Ethernet sub-interfaces.

  `Gig.*0/0/0/[1-5]|Gig.*0/0/0/[10-20]`—To match on either Gig.*0/0/0/[1-5] or Gig.*0/0/0/[10-20].

  `'TenGigE.*|POS.*`—To match on either TenGigE.* or POS.* .

- Commands that require a node identifier for the **location** keyword are not supported. For example, this configuration is not supported:

```
lpts pifib hardware police location 0/0/CPU0
```

- Overlapping regular expressions within a configuration group for the same configuration are not supported. For example:

```
group G-INTERFACE
interface 'gig.*a.*'
  mtu 1500
!
interface 'gig.*e.* '
  mtu 2000
!
end-group

interface gigabitethernet0/4/1/0
  apply-group G-INTERFACE
```

  This configuration is not permitted because it cannot be determined whether the `interface gigabitethernet0/4/1/0` configuration inherits `mtu 1500` or `mtu 2000`. Both expressions in the configuration group match gigabitethernet0/4/1/0.

- Up to eight configuration groups are permitted on one apply-group command.

- Use multi-line configuration style to configure Flexible CLI configuration groups (like group or apply-group commands) by entering each configuration mode in a separate line, one configuration per line. This is important so that the configuration properties are fully inherited and for better readability during troubleshooting.

  Example for a correct configuration style is:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router isis IGP
RP/0/RP0/CPU0:router(config-isis)# interface Ten 0/4/0/0
RP/0/RP0/CPU0:router(config-isis-if) # address-family ipv4 unicast
RP/0/RP0/CPU0:router (config-isis-if-af) # metric 123
```

Configuring a Configuration Group

A configuration group includes a series of configuration statements that can be used in multiple hierarchical levels in the router configuration tree. By using regular expressions in a configuration group, you can create generic commands that can be applied in multiple instances.

Use this task to create and use a configuration group.

Note Flexible CLI configurations are not available through the XML interface.

SUMMARY STEPS

1. **configure**
2. **group** *group-name*
3. Enter configuration commands, starting from global configuration mode. Use regular expressions for interface names and other variable instances.
4. **end-group**
5. **apply-group**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**group** *group-name* **Example:** `RP/0/RP0/CPU0:router(config)# group g-interf`	Specifies a name for a configuration group and enters group configuration mode to define the group.The *group-name* argument can have up to 32 characters and cannot contain any special characters. For information regarding special characters, refer to the *Understanding Regular Expressions, Special Characters, and Patterns* module in the *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*.

Step 3	Enter configuration commands, starting from global configuration mode. Use regular expressions for interface names and other variable instances.
	Example:
	`RP/0/RP0/CPU0:router(config)# group g-interf`
	`RP/0/RP0/CPU0:router(config-GRP)# interface 'GigabitEthernet.*'`
	`RP/0/RP0/CPU0:router(config-GRP-if)# mtu 1500`
		Specifies the configuration statements that you want included in this configuration group.
		For more information regarding the use of regular expressions, see Regular Expressions in Configuration Groups, on page 184. This example is applicable to all Gigabit Ethernet interfaces.

Step 4	**end-group** **Example:** `RP/0/RP0/CPU0:router(config-GRP-if)# end-group`	Completes the configuration of a configuration group and exits to global configuration mode.

Step 5	**apply-group**
	Example:
	`RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/2/0/0`
	`RP/0/RP0/CPU0:router(config-if)# apply-group`
		Adds the configuration of the configuration group into the router configuration applicable at the location that the group is applied. Groups can be applied in multiple locations, and their effect depends on the location and context.
		The MTU value from the group g-interf is applied to the interface GigabitEthernet0/2/0/0. If this group is applied in

	Command or Action	Purpose
	`g-interf`	global configuration mode, the MTU value is inherited by all Gigabit Ethernet interfaces that do not have an MTU value configured.

Simple Configuration Group: Example

This example shows how to use configuration groups to add a global configuration to the system:

```
RP/0/RP0/CPU0:router(config)# group g-logging
RP/0/RP0/CPU0:router(config-GRP)# logging trap notifications
RP/0/RP0/CPU0:router(config-GRP)# logging console debugging
RP/0/RP0/CPU0:router(config-GRP)# logging monitor debugging
RP/0/RP0/CPU0:router(config-GRP)# logging buffered 10000000
RP/0/RP0/CPU0:router(config-GRP)# end-group

RP/0/RP0/CPU0:router(config)# apply-group g-logging
```

When this configuration is committed, all commands contained in the g-logging configuration group are committed.

Configuration Group Applied to Different Places: Example

Configuration groups can be applied to different places, and their effect depends on the context within which they are applied. Consider this configuration group:

```
RP/0/RP0/CPU0:router(config)# group g-interfaces
RP/0/RP0/CPU0:router(config-GRP)# interface 'FastEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 1500
RP/0/RP0/CPU0:router(config-GRP-if)# exit
RP/0/RP0/CPU0:router(config-GRP)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 1000
RP/0/RP0/CPU0:router(config-GRP-if)# exit
RP/0/RP0/CPU0:router(config-GRP)# interface 'POS.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 2000
RP/0/RP0/CPU0:router(config-GRP-if)# end-group
```

This group can be applied to Fast Ethernet, Gigabit Ethernet or POS interfaces, and in each instance the applicable MTU is applied. For instance, in this example, the Gigabit Ethernet interface is configured to have an MTU of 1000:

```
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet0/2/0/0
RP/0/RP0/CPU0:router(config-if)# apply-group g-interfaces
RP/0/RP0/CPU0:router(config-if)# ipv4 address 2.2.2.2 255.255.255.0
```

In this example, the Fast Ethernet interface is configured to have an MTU of 1500:

```
RP/0/RP0/CPU0:router(config)# interface FastEthernet0/2/0/0
RP/0/RP0/CPU0:router(config-if)# apply-group g-interfaces
```

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 3.3.3.3 255.255.255.0
```

The same configuration group is used in both cases, but only the applicable configuration statements are used.

Verifying the Configuration of Configuration Groups

Use this task to verify the router configuration using configuration groups:

SUMMARY STEPS

1. **show running-config group** [*group-name*]
2. **show running-config**
3. **show running-config inheritance**
4. **show running-config interface x/y/z inheritance** *config-command*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show running-config group** [*group-name*]
	Example:
	`RP/0/RP0/CPU0:router# show running-config group`
	`group g-int-ge`
	` interface 'GigabitEthernet.*'`
	`  mtu 1000`
	`  negotiation auto`
	` !`
	`end-group`
		Displays the contents of a specific or all configured configuration groups.

Step 2	**show running-config**
	Example:
	`RP/0/RP0/CPU0:router# show running-config`
	`group G-INTERFACE-MTU`
	` interface 'POS.*'`
	`  mtu 1500`
	` !`
	`end-group`
	`interface POS0/4/1/0`
	` apply-group G-INTERFACE-MTU`
	`!`
	`interface POS0/4/1/1`
	` apply-group G-INTERFACE-MTU`
	` mtu 2000`
	`!`
		Displays the running configuration. Any applied groups are displayed. There is no indication as to whether these configuration groups affect the actual configuration or not. In this example, although the group G-INTERFACE-MTU is applied to POS0/4/1/1, the configured MTU value is 2000 and not 1500. This happens if the command **mtu 2000** is configured directly on the interface. An actual configuration overrides a configuration group configuration if they are the same.

	Command or Action	Purpose

Step 3	**show running-config inheritance**
	Example:
	`RP/0/RP0/CPU0:router# show running-config inheritance`
	`.`
	`.`
	`group G-INTERFACE-MTU`
	` interface 'POS.*'`
	`  mtu 1500`
	` !`
	`end-group`
	`.`
	`.`
	`interface POS0/4/1/0`
	` ## Inherited from group G-INTERFACE-MTU`
	` mtu 1500`
	`!`
	`interface POS0/4/1/1`
	` mtu 2000`
	`!`
	`.`
	`.`
		Displays the inherited configuration where ever a configuration group has been applied.

Step 4	**show running-config interface x/y/z inheritance** *config-command*
	Example:
	`RP/0/RP0/CPU0:router# show running-config interface pos0/4/1/0 inheritance [detail]`
	`interface POS0/4/1/0`
	` ## Inherited from group G-INTERFACE-MTU`
	` mtu 1500`
		Displays the inherited configuration for a specific configuration command.

Apply Groups Priority Inheritance

The inheritance is supported according to the priority.

Apply groups priority inheritance helps flexible configuration groups handle common configuration statements between groups. When multiple configuration groups have common configuration statements, the inheritance priority is configuration statements present in inner groups have precedence over configuration statements present in outer groups. Tiebreaker is determined by the system order (lexicographical) of the regular expressions. User defined order of commands are not accepted.

For example, a configuration statement in configuration group ONE has precedence over any other group. A configuration statement in configuration group SEVEN is used only if it is not contained in any other group. Within a configuration group, inheritance priority is lengthiest match.

```
apply-group SIX SEVEN
    router ospf 0
    apply-group FOUR FIVE
```

```
   area 0
    apply-group THREE
    interface GigabitEthernet 0/0/0/0
      apply-group ONE TWO
  !
 !
!
```

The above example states two scenarios. Inner most group (**apply-group ONE TWO**) has the highest priority.
Case 1

In the first scenario it shows which group gets the first priority. The example states which group is applied between different configuration groups (different groups- nothing in common between them). While applying the group one (ONE TWO), all the seven groups that matches to the interface `interface GigabitEthernet 0/0/0/0` will be applied.

Case 2

In the case when all these groups (mentioned above) have same (common) configuration, group one will be active. The `apply-group ONE TWO` will be active. If group ONE is deleted then group TWO will be active.

Regular Expressions in Configuration Groups

Regular expressions are used in configuration groups to make them widely applicable. Portable Operating System Interface for UNIX (POSIX) 1003.2 regular expressions are supported in the names of configuration statements. Single quotes must be used to delimit a regular expression.

For general information regarding regular expressions, refer to the *Understanding Regular Expressions, Special Characters, and Patterns* module in the *Cisco IOS XR Getting Started Guide for the Cisco CRS Router*.

Note Not all POSIX regular expressions are supported. Refer to Flexible Configuration Restrictions, on page 178 for more information.

Regular Expressions for Interface Identifiers

Configuration groups do not accept exact interface identifiers. You must use a regular expression to identify a group of interfaces that are applicable to the configuration group. The regular expression '.*' is not allowed. You must begin the regular expression for an interface identifier with an unambiguous word, followed by the regular expression. For example, to configure Gigabit Ethernet interfaces, use the regular expression 'GigabitEthernet.*'.

To display a list of available interface types for your router configuration, enter **interface ?** at the configuration group prompt:

```
RP/0/RP0/CPU0:router(config-GRP)# interface ?

  ATM              'RegExp': ATM Network Interface(s)
  BVI              'RegExp': Bridge-Group Virtual Interface
  Bundle-Ether     'RegExp': Aggregated Ethernet interface(s)
  Bundle-POS       'RegExp': Aggregated POS interface(s)
  GigabitEthernet  'RegExp': GigabitEthernet/IEEE 802.3 interface(s)
  IMA              'RegExp': ATM Network Interface(s)
  Loopback         'RegExp': Loopback interface(s)
```

```
  MgmtEth          'RegExp': Ethernet/IEEE 802.3 interface(s)
  Multilink        'RegExp': Multilink network interface(s)
  Null             'RegExp': Null interface
  POS              'RegExp': Packet over SONET/SDH network interface(s)
  PW-Ether         'RegExp': PWHE Ethernet Interface
  PW-IW            'RegExp': PWHE VC11 IP Interworking Interface
  Serial           'RegExp': Serial network interface(s)
  tunnel-ip        'RegExp': GRE/IPinIP Tunnel Interface(s)
  tunnel-mte       'RegExp': MPLS Traffic Engineering P2MP Tunnel interface(s)
  tunnel-te        'RegExp': MPLS Traffic Engineering Tunnel interface(s)
  tunnel-tp        'RegExp': MPLS Transport Protocol Tunnel interface
```

Note Although you are required to enter only enough characters for the interface type to be unique, it is recommended that you enter the entire phrase. All interface types used in regular expressions are case-sensitive.

To specify a subinterface, prefix the expression with the characters \. (backslash period). For example, use `interface 'GigabitEthernet.*\..*'` to configure all Gigabit Ethernet subinterfaces.

You can specify Layer 2 transport interfaces or point-to-point interfaces as shown in these examples:

```
group g-l2t
  interface 'Gi.*\..*' l2transport
.
.
end-group
group g-ptp
  interface 'Gi.*\..*' point-to-point
.
.
end-group
```

Regular Expressions for an OSPF Configuration

Exact router process names and OSPF areas cannot be used. You must use a regular expression to specify a process name or group of OSPF areas. To specify that the OSFP area can be either a scalar value or an IP address, use the regular expression '.*', as in this example:

```
group g-ospf
router ospf '.*'
area '.*'
mtu-ignore enable
!
!
end-group
```

To specify that the OSPF area must be an IP address, use the expression '\.' as in this example:

```
group g-ospf-ipaddress
router ospf '.*\..*\..*\..*'
area '.*'
passive enable
!
!
 end-group
```

To specify that the OSPF area must be a scalar value, use the expression '1.*', as in this example:

```
group g-ospf-match-number
router ospf '.*'
area '1.*'
passive enable
!
!
end-group
```

Regular Expressions for a BGP AS

Exact BGP AS values cannot be used in configuration groups. Use a regular expression to specify either AS plain format, or AS dot format as in the format X.Y. To match AS plain format instances, use a simple regular expression. To match AS dot format instances, use two regular expressions separated by a dot, as shown in this example:

```
group g-bgp
router bgp '*'.'*'
address-family ipv4 unicast
!
!
end-group
```

Regular Expressions for ANCP

Exact Access Node Control Protocol (ANCP) sender-name identifiers cannot be used in configuration groups. Because the sender name argument can be either an IP address or a MAC address, you must specify in the regular expression which one is being used. Specify an IP address as `'.*\..*\..*\..*'`; specify a MAC address as `'.*\..*\..*'`.

Resolving to a Uniform Type

Regular expressions must resolve to a uniform type. This is an example of an illegal regular expression:

```
group g-invalid
 interface '.*'
  bundle port-priority 10
 !
interface '.*Ethernet.*'
  bundle port-priority 10
 !
end-group
```

In this example, the **bundle** command is supported for interface type GigabitEthernet but not for interface type 'FastEthernet'. The regular expressions '.*' and '.*Ethernet.*' match both GigabitEthernet and FastEthernet types. Because the **bundle** command is not applicable to both these interface types, they do not resolve to a uniform type and therefore the system does not allow this configuration.

Note If the system cannot determine from the regular expression what the configuration should be, the expression is not considered valid.

Note The regular expression '.*' is not allowed when referring to an interface identifier. You must begin the regular expression for an interface identifier with an unambiguous word, followed by the regular expression. Refer to *Regular Expressions for Interface Identifiers* in this section for more information.

Overlapping Regular Expressions

Regular expressions are used in names of configuration statements within a configuration group. This permits inheritance by the configuration when applied to matching names. Single quotes are used to delimit the regular expression. Overlapping regular expression within a configuration group for the same configuration is permitted.

The example, given below, illustrates the process of creating and applying multiple configuration groups:

```
RP/0/RP0/CPU0:router(config)#group FB_flexi_snmp
RP/0/RP0/CPU0:router(config-GRP)# snmp-server vrf '.*'
RP/0/RP0/CPU0:router(config-GRP-snmp-vrf)# host 1.1.1.1 traps version 2c group_1
RP/0/RP0/CPU0:router(config-GRP-snmp-vrf)# host 1.1.1.1 informs version 2c group_1
RP/0/RP0/CPU0:router(config-GRP-snmp-vrf)# context group_1

RP/0/RP0/CPU0:router(config-GRP-snmp-vrf)#
RP/0/RP0/CPU0:router(config-GRP-snmp-vrf)#commit

RP/0/RP0/CPU0:router(config-GRP-snmp-vrf)#root
RP/0/RP0/CPU0:router(config)#
RP/0/RP0/CPU0:router(config)#snmp-server vrf vrf1
RP/0/RP0/CPU0:router(config-snmp-vrf)#snmp-server vrf vrf10
RP/0/RP0/CPU0:router(config-snmp-vrf)#!
RP/0/RP0/CPU0:router(config-snmp-vrf)#snmp-server vrf vrf100
RP/0/RP0/CPU0:router(config-snmp-vrf)#
RP/0/RP0/CPU0:router(config-snmp-vrf)#commit

RP/0/RP0/CPU0:router(config-snmp-vrf)#root
RP/0/RP0/CPU0:router(config)#
RP/0/RP0/CPU0:router(config)#apply-group FB_flexi_snmp
RP/0/RP0/CPU0:router(config)#do sh running-config group
group FB_flexi_snmp
 snmp-server vrf '.*'
  host 1.1.1.1 traps version 2c group_1
  host 1.1.1.1 informs version 2c group_1
  context group_1
 !
end-group
apply-group FB_flexi_snmp
snmp-server vrf vrf1
!
snmp-server vrf vrf10
!
snmp-server vrf vrf100
!
RP/0/0/CPU0:ios#show running-config inheritance detail


group FB_flexi_snmp
 snmp-server vrf '.*'
  host 1.1.1.1 traps version 2c group_1
  host 1.1.1.1 informs version 2c group_1
  context group_1
 !
```

```
end-group
snmp-server vrf vrf1
 ## Inherited from group FB_flexi_snmp
 host 1.1.1.1 traps version 2c group_1
 ## Inherited from group FB_flexi_snmp
 host 1.1.1.1 informs version 2c group_1
 ## Inherited from group FB_flexi_snmp
 context group_1
!
snmp-server vrf vrf10
 ## Inherited from group FB_flexi_snmp
 host 1.1.1.1 traps version 2c group_1
 ## Inherited from group FB_flexi_snmp
 host 1.1.1.1 informs version 2c group_1
 ## Inherited from group FB_flexi_snmp
 context group_1
!
snmp-server vrf vrf100
 ## Inherited from group FB_flexi_snmp
 host 1.1.1.1 traps version 2c group_1
 ## Inherited from group FB_flexi_snmp
 host 1.1.1.1 informs version 2c group_1
 ## Inherited from group FB_flexi_snmp
 context group_1
```

The example given below demonstrates the regular expression. In this example `snmp-server vrf '.*'` and `snmp-server vrf '[\w]+` are two different regular expressions.

```
group FB_flexi_snmp
snmp-server vrf '.*'
host 1.1.1.1 traps version 2c group_1
host 1.1.1.1 informs version 2c group_1
context group_1
!
snmp-server vrf '[\w]+'
host 2.2.2.2 traps version 2c group_2
host 2.2.2.2 informs version 2c group_2
context group_2
!
end-group
```

This individual regular expression gets combined to all the three expressions - `snmp-server vrf vrf1`, `snmp-server vrf vrf10` and `snmp-server vrf vrf100` as given below.

```
apply-group FB_flexi_snmp
snmp-server vrf vrf1
!
snmp-server vrf vrf10
!
snmp-server vrf vrf100
!
```

In a configuration group, there can be instances of regular expressions overlap. In such cases, the regular expression with the highest priority is activated and inherited, when applied. It has that regular expression, which comes first in the lexicographic order that has the highest priority.

The following example shows how to use overlapping regular expressions and how the expression with higher priority is applied:

```
 group FB_flexi_snmp

 snmp-server vrf '.*'

  host 1.1.1.1 traps version 2c group_1

  host 1.1.1.1 informs version 2c group_1

  context group_1

 !

 snmp-server vrf '[\w]+'

  host 2.2.2.2 traps version 2c group_2
  host 2.2.2.2 informs version 2c group_2
  context group_2
 !
end-group
```

The expression shown below has the highest priority:

```
  group FB_flexi_snmp

  snmp-server vrf '.*'

  host 1.1.1.1 traps version 2c group_1

  host 1.1.1.1 informs version 2c group_1

  context group_1
```

The examples given above, show two different regular expression `snmp-server vrf '.*'` and `snmp-server vrf '[\w]+'`.

The expression below, shows how these two expressions get merged together:

```
 apply-group FB_flexi_snmp

 snmp-server vrf vrf1
 !
 snmp-server vrf vrf10
 !
 snmp-server vrf vrf100
 !
```

Any change in a regular expression with lower priority will not affect the inheritance.

Any changes made to an existing regular expression, which is of less (non-top) priority, it will not have any effect on the inheritance.

```
 snmp-server vrf '[\w]+'

 host 2.2.2.2 traps version 2c group_2
 host 2.2.2.2 informs version 2c group_2
 context group_2
```

The expression with the higher priority gets inherited, as shown below:

```
group FB_flexi_snmp

snmp-server vrf '.*'

 host 1.1.1.1 traps version 2c group_1

 host 1.1.1.1 informs version 2c group_1

 context group_1
```

Apply Groups Priority Inheritance

Priority governs inheritance.

Note From the Release 6.3.1 onwards, you are able to enter the Flexible CLI config group definition, **apply-group** and **exclude-group** command in any order as long as the entire commit has all the group definitions needed.

Apply groups priority inheritance helps flexible configuration groups to handle common configuration statements between groups. When multiple configuration groups have common configuration statements, the inheritance priority is such that the configuration statements present in inner groups have precedence over those configuration statements present in outer groups. In case of tiebreakers, the priority is assigned in accordance to the lexicographical order of regular expressions. User defined order of commands are not accepted.

For example, a configuration statement in configuration group ONE has precedence over another group. A configuration statement in configuration group SEVEN is used only if it does not exist in any other group. Within a configuration group, inheritance priority is the longest match.

```
apply-group SIX SEVEN
    router ospf 0
    apply-group FOUR FIVE
   area 0
    apply-group THREE
    interface GigabitEthernet 0/0/0/0
      apply-group ONE TWO
  !
 !
!
```

The above example shows two scenarios. The inner most group (**apply-group ONE TWO**) has the highest priority. Case 1

The first scenario shows which group gets the priority. The example states which group is applied between different configuration groups (different groups with nothing in common). While applying group one (ONE TWO), all the seven groups matches the interface `interface GigabitEthernet 0/0/0/0`- is applied.

Case 2

Here, when all have the same (common) configuration, group one will be active. That is `apply-group ONE TWO` is active. If group ONE is deleted, then group TWO will be active.

Configuration Examples Using Regular Expressions

Configuration Group with Regular Expression: Example

This example shows the definition of a configuration group for configuring Gigabit Ethernet interfaces with ISIS routing parameters, using regular expressions for the exact interface:

```
RP/0/RP0/CPU0:router(config)# group g-isis-gige
RP/0/RP0/CPU0:router(config-GRP)# router isis '.*'
RP/0/RP0/CPU0:router(config-GRP-isis)# interface 'GigabitEthernet.*'
RP/0/RP0/CPU0:router(config-GRP-isis-if)# lsp-interval 20
RP/0/RP0/CPU0:router(config-GRP-isis-if)# hello-interval 40
RP/0/RP0/CPU0:router(config-GRP-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-GRP-isis-if-af)# metric 10
RP/0/RP0/CPU0:router(config-GRP-isis-if-af)# end-group
RP/0/RP0/CPU0:router(config)#
```

To illustrate the use of this configuration group, assume that you want to configure these Gigabit Ethernet interfaces with the ISIS routing parameters:

```
router isis green
 interface GigabitEthernet0/0/0/0
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
   metric 10
  !
 !
 interface GigabitEthernet0/0/0/1
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
   metric 10
  !
 !
 interface GigabitEthernet0/0/0/2
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
   metric 10
  !
 !
 interface GigabitEthernet0/0/0/3
  lsp-interval 20
  hello-interval 40
  address-family ipv4 unicast
   metric 10
  !
 !
!
```

There are three possible ways to use the configuration group to configure these interfaces. The first is by applying the group within the interface configuration, as shown here:

```
router isis green
 interface GigabitEthernet0/0/0/0
  apply-group g-isis-gige
  !
 !
 interface GigabitEthernet0/0/0/1
  apply-group g-isis-gige
  !
 !
 interface GigabitEthernet0/0/0/2
  apply-group g-isis-gige
  !
 !
 interface GigabitEthernet0/0/0/3
  apply-group g-isis-gige
  !
 !
```

In this situation, only the interfaces to which you apply the configuration group inherit the configuration.

The second way to configure these interfaces using the configuration group is to apply the configuration group within the **router isis** configuration, as shown here:

```
router isis green
 apply-group g-isis-gige
 interface GigabitEthernet0/0/0/0
 !
 interface GigabitEthernet0/0/0/1
 !
 interface GigabitEthernet0/0/0/2
 !
 interface GigabitEthernet0/0/0/3
 !
!
```

In this way, any other Gigabit Ethernet interfaces that you configure in the ISIS green configuration also inherit these configurations.

The third way to configure these interfaces using the configuration group is to apply the group at the global level as shown here:

```
apply-group g-isis-gige
router isis green
 interface GigabitEthernet0/0/0/0
 !
 interface GigabitEthernet0/0/0/1
 !
 interface GigabitEthernet0/0/0/2
 !
 interface GigabitEthernet0/0/0/3
 !
!
```

In this example, the configuration of the group is applied to all Gigabit Ethernet interfaces configured for ISIS.

Configuration Group Inheritance with Regular Expressions: Example

Local Configuration Has Precedence Over Configuration Group

An explicit configuration takes precedence over a configuration applied from a configuration group. For example, assume that this configuration is running on the router:

```
router ospf 100
 packet-size 1000
!
```

You configure this configuration group, apply it, and commit it to the configuration.

```
RP/0/RP0/CPU0:router(config)# group g-ospf
RP/0/RP0/CPU0:router(config-GRP)# router ospf '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf)# nsf cisco
RP/0/RP0/CPU0:router(config-GRP-ospf)# packet-size 3000
RP/0/RP0/CPU0:router(config-GRP-ospf)# end-group

RP/0/RP0/CPU0:router(config)# apply-group g-ospf
```

The result is effectively this configuration:

```
router ospf 100
 packet-size 1000
 nsf cisco
```

Note that `packet-size 3000` is not inherited from the configuration group because the explicit local configuration has precedence.

Compatible Configuration Is Inherited

The configuration in the configuration group must match the configuration on the router to be inherited. If the configuration does not match, it is not inherited. For example, assume that this configuration is running on the router:

```
router ospf 100
 auto-cost disable
!
```

You configure this configuration and commit it to the configuration.

```
RP/0/RP0/CPU0:router(config)# group g-ospf
RP/0/RP0/CPU0:router(config-GRP)# router ospf '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf)# area '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# packet-size 2000
RP/0/RP0/CPU0:router(config-GRP-ospf)# end-group

RP/0/RP0/CPU0:router(config)# apply-group g-ospf

RP/0/RP0/CPU0:router(config)# router ospf 200
RP/0/RP0/CPU0:router(config-ospf)# area 1
```

The result is effectively this configuration:

```
router ospf 100
  auto-cost disable

router ospf 200
  area 1
  packet-size 2000
```

The packet size is inherited by the ospf 200 configuration, but not by the ospf 100 configuration because the area is not configured.

Layer 2 Transport Configuration Group: Example

This example shows how to configure and apply a configuration group with Layer 2 transport subinterfaces:

```
RP/0/RP0/CPU0:router(config)# group g-l2trans-if
RP/0/RP0/CPU0:router(config-GRP)# interface 'TenGigE.*\..*' l2transport
RP/0/RP0/CPU0:router(config-GRP)# mtu 1514
RP/0/RP0/CPU0:router(config-GRP)# end-group

RP/0/RP0/CPU0:router(config)# interface TenGigE0/0/0/0.1 l2transport
RP/0/RP0/CPU0:router(config-if)# apply-group g-l2trans-if
```

When this configuration is committed, the Ten Gigabit Ethernet interface 0/0/0/0.1 inherits the 1514 MTU value. This is the output displayed from the **show running-config inheritence** command for the Ten Gigabit Ethernet interface:

```
interface TenGigE0/0/0/0.1 l2transport
 ## Inherited from group g-l2trans-if
 mtu 1514
!
```

Configuration Group Precedence: Example

When similar configuration statements are contained in multiple configuration groups, groups applied in inner configuration modes take precedence over groups applied in outer modes. This example shows two configuration groups that configure different cost values for OSPF.

```
RP/0/RP0/CPU0:router(config)# group g-ospf2
RP/0/RP0/CPU0:router(config-GRP)# router ospf '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf)# area '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# cost 2
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# end-group

RP/0/RP0/CPU0:router(config)# group g-ospf100
RP/0/RP0/CPU0:router(config-GRP)# router ospf '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf)# area '.*'
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# cost 100
RP/0/RP0/CPU0:router(config-GRP-ospf-ar)# end-group
```

If these configuration groups are applied as follows, the cost 2 specified in g-ospf2 is inherited by OSPF area 0 because the group is applied in a more inner configuration mode. In this case, the configuration in group g-ospf100 is ignored.

```
RP/0/RP0/CPU0:router(config)# router ospf 0
RP/0/RP0/CPU0:router(config-ospf)# apply-group g-ospf100
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# apply-group g-ospf2
```

Changes to Configuration Group are Automatically Inherited: Example

When you make changes to a configuration group that is committed and applied to your router configuration, the changes are automatically inherited by the router configuration. For example, assume that this configuration is committed:

```
group g-interface-mtu
 interface 'POS.*'
  mtu 1500
 !
end-group

interface POS0/4/1/0
 apply-group g-interface-mtu
!
```

Now you change the configuration group as in this example:

```
RP/0/RP0/CPU0:router(config)# group g-interface-mtu
RP/0/RP0/CPU0:router(config-GRP)# interface 'POS.*'
RP/0/RP0/CPU0:router(config-GRP-if)# mtu 2000
RP/0/RP0/CPU0:router(config-GRP-if)# end-group
```

When this configuration group is committed, the MTU configuration for interface POS0/4/1/0 is automatically updated to 2000.

Configuration Examples for Flexible CLI Configuration

Basic Flexible CLI Configuration: Example

This example shows that the Media Access Control (MAC) accounting configuration from the gd21 configuration group is applied to all Gigabit Ethernet interfaces in slot 2, ports 1 to 9.

1. Configure the configuration group that configures MAC accounting:

```
RP/0/RP0/CPU0:router# show running group gd21

group gd21
interface 'GigabitEthernet0/0/0/2[1-9]'
description general interface inheritance check
```

```
load-interval 30
mac-accounting ingress
mac-accounting egress
!
end-group
```

2. Check that the corresponding apply-group is configured in global configuration or somewhere in the hierarchy:

```
RP/0/RP0/CPU0:router# show running | in apply-group gd21

Building configuration...
apply-group gd21
```

3. Check the concise local view of the configuration of some of the interfaces:

```
RP/0/RP0/CPU0:router# show running interface

interface GigabiEthernet0/0/0/21
!
interface GigabitEthernet0/0/0/22
!
```

4. Verify that the match and inheritance occur on these interfaces:

```
RP/0/RP0/CPU0:router# show running inheritance interface

interface GigabitEthernet0/0/0/21
## Inherited from group gd21
description general interface inheritance check
## Inherited from group gd21
load-interval 30
## Inherited from group gd21
mac-accounting ingress
## Inherited from group gd21
mac-accounting egress
!
Interface GigabitEthernet0/0/0/22
## Inherited from group gd21
description general interface inheritance check
## Inherited from group gd21
load-interval 30
## Inherited from group gd21
mac-accounting ingress
## Inherited from group gd21
mac-accounting egress
!
!
```

5. Verify that the inherited configuration actually takes effect:

```
RP/0/RP0/CPU0:router# show mac gigabitEthernet0/0/0/21

GigabitEthernet0/0/0/21
  Input (96 free)
    6c9c.ed35.90fd:  1271 packets, 98426 bytes
        Total: 1271 packets, 98426 bytes
  Output (96 free)
    6c9c.ed35.90fd:   774 packets, 63265 bytes
```

```
Total: 774 packets, 63264 bytes
```

Interface MTU Settings for Different Interface Types: Example

This example shows that an MTU value is configured on different interface types.

1. Configure an interface MTU configuration group and apply this group:

```
RP/0/RP0/CPU0:router# show running group l2tr

group l2tr
interface 'GigabitEthernet0/0/0/3.*'
mtu 1500
!
interface 'GigabitEthernet0/0/0/9\..*'
mtu 1400
!
interface 'GigabitEthernet0/0/0/9\..*' l2transport
mtu 1400
!
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...

apply-group l2tr
```

2. Check the concise view and the inheritance view of the various interfaces:

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/30

interface GigabitEthernet0/0/0/30
!
RP/0/RP0/CPU0:router# show running inheritance interface gigabitEthernet0/0/0/30

interface GigabitEthernet0/0/0/30
## Inherited from group l2tr
mtu 1500
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/9.800

interface GigabitEthernet0/0/0/9.800
  encapsulation dot1q 800
!

RP/0/RP0/CPU0:router# show running inheritance interface gigabitEthernet0/0/0/9.800

interface GigabitEthernet0/0/0/9.800
## Inherited from group l2tr
mtu 1400
encapsulation dot1q800
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet0/0/0/9.250
```

```
interface GigabitEthernet0/0/0/9.250 l2transport
  encapsulation dot1q 250
!

RP/0/RP0/CPU0:router# show running inheritance interface gigabitEthernet0/0/0/9.800

interface GigabitEthernet0/0/0/9.250 l2transport
encapsulation dot1q250
## Inherited from group l2tr
mtu 1400
!
```

3. Verify that the correct values from the group do take effect:

```
RP/0/RP0/CPU0:router# show interface gigabitEthernet 0/0/0/30

GigabitEthernet0/0/0/30 is down, line protocol is down
  Interface state transitions: 0
  Hardware is GigabitEthernet, address is 0026.9824.ee56 (bia 0026.9824.ee56)
  Internet address is Unknown
  MTU 1500 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
     reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 1000Mb/s, link type is force-up
  output flow control is off, input flow control is off
  loopback not set,
  Last input never, output never
  Last clearing of "show interface" counters never
  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 total input drops
     0 drops for unrecognized upper-level protocol
     Received 0 broadcast packets, 0 multicast packets
              0 runts, 0 giants, 0 throttles, 0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 total output drops
     Output 0 broadcast packets, 0 multicast packets
     0 output errors, 0 underruns, 0 applique, 0 resets
     0 output buffer failures, 0 output buffers swapped out

RP/0/RP0/CPU0:router# show interface gigabitEthernet 0/0/0/9.801

GigabitEthernet0/0/0/9.801 is up, line protocol is up
  Interface state transitions: 1
  Hardware is VLAN sub-interface(s), address is 0026.9824.ee41
  Internet address is Unknown
  MTU 1400 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
     reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation 802.1Q Virtual LAN, VLAN Id 801,  loopback not set,
  Last input never, output never
  Last clearing of "show interface" counters never
  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 total input drops
     0 drops for unrecognized upper-level protocol
     Received 0 broadcast packets, 0 multicast packets
     0 packets output, 0 bytes, 0 total output drops
     Output 0 broadcast packets, 0 multicast packets


RP/0/RP0/CPU0:router# show interface gigabitEthernet 0/0/0/9.250
```

```
GigabitEthernet0/0/0/9.250 is up, line protocol is up
  Interface state transitions: 1
  Hardware is VLAN sub-interface(s), address is 0026.9824.ee41
  Layer 2 Transport Mode
  MTU 1400 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
     reliability Unknown, txload Unknown, rxload Unknown
  Encapsulation 802.1Q Virtual LAN,
    Outer Match: Dot1Q VLAN 250
    Ethertype Any, MAC Match src any, dest any
  loopback not set,
  Last input never, output never
  Last clearing of "show interface" counters never
     0 packets input, 0 bytes
     0 input drops, 0 queue drops, 0 input errors
     0 packets output, 0 bytes

     0 output drops, 0 queue drops, 0 output errors
```

ACL Referencing: Example

This example shows how to reference access-lists on a number of interfaces using configuration groups.

1. Configure the configuration group and apply-group:

```
RP/0/RP0/CPU0:router# show running group acref

group acref
 interface 'GigabitEthernet0/0/0/3.*'
  ipv4 access-group adem ingress
  ipv4 access-group adem egress
 !
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...

apply-group isis l2tr isis2 mpp bundle1 acref
```

2. Check the concise and inheritance view of the matching configurations:

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/30

interface GigabitEthernet0/0/0/30
!

RP/0/RP0/CPU0:router# show running inheritance interface GigabitEthernet 0/0/0/30

interface GigabitEthernet0/0/0/30
 ## Inherited from group l2tr
 mtu 1500
 ## Inherited from group acref
 ipv4 access-group adem ingress
 ## Inherited from group acref
 ipv4 access-group adem egress
!
```

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/31

interface GigabitEthernet0/0/0/31
!

RP/0/RP0/CPU0:router# show running inheritance interface GigabitEthernet 0/0/0/31

interface GigabitEthernet0/0/0/31
 ## Inherited from group l2tr
 mtu 1500
 ## Inherited from group acref
 ipv4 access-group adem ingress
 ## Inherited from group acref
 ipv4 access-group adem egress
```

3. Check that the ACL group configuration actually got configured by using a traffic generator and watching that denied traffic is dropped.

Local Configuration Takes Precedence: Example

This example illustrates that local configurations take precedence when there is a discrepancy between a local configuration and the configuration inherited from a configuration group.

1. Configure a local configuration in a configuration submode with an access list:

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/39

interface GigabitEthernet0/0/0/39
 ipv4 access-group smany ingress
 ipv4 access-group smany egress
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/38

interface GigabitEthernet0/0/0/38
!

RP/0/RP0/CPU0:router# show running ipv4 access-list smany

ipv4 access-list smany
 10 permit ipv4 any any
!

RP/0/RP0/CPU0:router# show running ipv4 access-list adem

ipv4 access-list adem
 10 permit ipv4 21.0.0.0 0.255.255.255 host 55.55.55.55
 20 deny ipv4 any any
!
```

2. Configure and apply the access list group configuration:

```
RP/0/RP0/CPU0:router# show running group acref

group acref
 interface 'GigabitEthernet0/0/0/3.*'
  ipv4 access-group adem ingress
```

```
  ipv4 access-group adem egress
 !
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...
apply-group isis l2tr isis2 mpp bundle1 acref
```

3. Check the concise and inheritance views for the matching interface where the access list reference is configured locally:

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/39

interface GigabitEthernet0/0/0/39
 ipv4 access-group smany ingress
 ipv4 access-group smany egress
!

RP/0/RP0/CPU0:router# show running inheritance interface gigabitEthernet 0/0/0/39

interface GigabitEthernet0/0/0/39
 ## Inherited from group l2tr
 mtu 1500
 ipv4 access-group smany ingress
 ipv4 access-group smany egress     << no config inherited, local config prioritized
!

RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/0/0/38

interface GigabitEthernet0/0/0/38
!

RP/0/RP0/CPU0:router# show running inheritance interface gigabitEthernet 0/0/0/38

interface GigabitEthernet0/0/0/38
 ## Inherited from group l2tr
 mtu 1500
 ## Inherited from group acref
 ipv4 access-group adem ingress
 ## Inherited from group acref
 ipv4 access-group adem egress
!
```

4. Use a traffic generator to verify that the traffic pattern for interface GigabitEthernet0/0/0/39 gets acted on by the access list in the local configuration (smany) and not according to the inherited referenced access list (adem).

ISIS Hierarchical Configuration: Example

This example illustrates inheritance and priority handling with two ISIS groups using an ISIS configuration.

1. Configure the local ISIS configuration:

```
RP/0/RP0/CPU0:router# show running router isis
```

```
router isis vink
 net 49.0011.2222.2222.2222.00
 address-family ipv4 unicast
  mpls traffic-eng level-1-2
  mpls traffic-eng router-id Loopback0
  redistribute connected
 !
 interface Bundle-Ether1
  address-family ipv4 unicast
  !
 !
 interface Bundle-Ether2
 !
 interface Loopback0
 !
 interface TenGigE0/2/0/0.3521
  address-family ipv4 unicast
  !
 !
 interface TenGigE0/2/0/0.3522
  address-family ipv4 unicast
  !
 !
 interface TenGigE0/2/0/0.3523
  address-family ipv4 unicast
  !
 !
 interface TenGigE0/2/0/0.3524
  address-family ipv4 unicast
  !
 !
 interface TenGigE0/2/0/0.3525
  address-family ipv4 unicast
  !
 !
 interface TenGigE0/2/0/0.3526
 !
 interface TenGigE0/2/0/0.3527
 !
 interface TenGigE0/2/0/0.3528
 !
 interface TenGigE0/2/0/1
  address-family ipv4 unicast
  !
 !
!
```

2. Configure two ISIS groups and apply these to the configuration:

```
RP/0/RP0/CPU0:router# show running group isis

group isis
 router isis '.*'
  address-family ipv4 unicast
   mpls traffic-eng level-1-2
   mpls traffic-eng router-id Loopback0
   redistribute connected
   redistribute ospf 1 level-1-2
  !
  interface 'TenGig.*'
   lsp-interval 40
```

```
   hello-interval 15
   address-family ipv4 unicast
    metric 50
   !
  !
  interface 'Bundle-Ether.*'
   address-family ipv4 unicast
    metric 55
   !
  !
 !
end-group

RP/0/RP0/CPU0:router# show running group isis2

group isis2
 router isis '.*'
 !
 router isis '^(vink)'
  address-family ipv4 unicast
  !
  interface '(^Ten)Gig.*'
  !
  interface '^(Ten)Gig.*'
   address-family ipv4 unicast
    metric 66
   !
  !
 !
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...

apply-group isis  l2tr  isis2  mpp  bundle1  acref
```

3. Check the inheritance view of the ISIS configuration:

```
RP/0/RP0/CPU0:router# show running inheritance router isis

router isis vink
 net 49.0011.2222.2222.2222.00
 address-family ipv4 unicast
  mpls traffic-eng level-1-2
  mpls traffic-eng router-id Loopback0
  redistribute connected
  ## Inherited from group isis
  redistribute ospf 1 level-1-2
 !
 interface Bundle-Ether1
  address-family ipv4 unicast
   ## Inherited from group isis
   metric 55
  !
 !
 interface Bundle-Ether2
  ## Inherited from group isis
  address-family ipv4 unicast
   ## Inherited from group isis
   metric 55
  !
```

```
!
interface Loopback0
!
interface TenGigE0/2/0/0.3521
 ## Inherited from group isis
 lsp-interval 40
 ## Inherited from group isis
 hello-interval 15
 address-family ipv4 unicast
  ## Inherited from group isis
  metric 50
 !
!
interface TenGigE0/2/0/0.3522
 ## Inherited from group isis
 lsp-interval 40
 ## Inherited from group isis
 hello-interval 15
 address-family ipv4 unicast
  ## Inherited from group isis
  metric 50
 !
!
interface TenGigE0/2/0/0.3523
 ## Inherited from group isis
 lsp-interval 40
 ## Inherited from group isis
 hello-interval 15
 address-family ipv4 unicast
  ## Inherited from group isis
  metric 50
 !
!
interface TenGigE0/2/0/0.3524
 ## Inherited from group isis
 lsp-interval 40
 ## Inherited from group isis
 hello-interval 15
 address-family ipv4 unicast
  ## Inherited from group isis
  metric 50
 !
!
interface TenGigE0/2/0/0.3525
 ## Inherited from group isis
 lsp-interval 40
 ## Inherited from group isis
 hello-interval 15
 address-family ipv4 unicast
  ## Inherited from group isis
  metric 50
 !
!
interface TenGigE0/2/0/0.3526
 ## Inherited from group isis
 lsp-interval 40
 ## Inherited from group isis
 hello-interval 15
 ## Inherited from group isis
 address-family ipv4 unicast
  ## Inherited from group isis
  metric 50
 !
!
```

```
 interface TenGigE0/2/0/0.3527
  ## Inherited from group isis
  lsp-interval 40
  ## Inherited from group isis
  hello-interval 15
  ## Inherited from group isis
  address-family ipv4 unicast
   ## Inherited from group isis
   metric 50
  !
 !
 interface TenGigE0/2/0/0.3528
  ## Inherited from group isis
  lsp-interval 40
  ## Inherited from group isis
  hello-interval 15
  ## Inherited from group isis
  address-family ipv4 unicast
   ## Inherited from group isis
   metric 50
  !
 !
 interface TenGigE0/2/0/1
  ## Inherited from group isis
  lsp-interval 40
  ## Inherited from group isis
  hello-interval 15
  address-family ipv4 unicast
   ## Inherited from group isis
   metric 50
  !
 !
!
```

4. Verify the actual functionality:

```
RP/0/RP0/CPU0:router# show isis interface TenGigE0/2/0/0.3528 | inc Metric

Metric (L1/L2):         50/50
```

OSPF Hierarchy: Example

This example illustrates hierarchical inheritance and priority. The configuration that is lower in hierarchy gets the highest priority.

1. Configure a local OSPF configuration:

```
RP/0/RP0/CPU0:router# show running router ospf

router ospf 1
 apply-group go-c
 nsr
 router-id 121.121.121.121
 nsf cisco
 redistribute connected
 address-family ipv4 unicast
 area 0
  apply-group go-b
```

```
  interface GigabitEthernet0/0/0/0
   apply-group go-a
  !
  interface GigabitEthernet0/0/0/1
  !
  interface GigabitEthernet0/0/0/3
  !
  interface GigabitEthernet0/0/0/4
  !
  interface GigabitEthernet0/0/0/21
   bfd minimum-interval 100
   bfd fast-detect
   bfd multiplier 3
  !
  interface TenGigE0/2/0/0.3891
  !
  interface TenGigE0/2/0/0.3892
  !
  interface TenGigE0/2/0/0.3893
  !
  interface TenGigE0/2/0/0.3894
  !
 !
!
router ospf 100
!
router ospf 1000
!
router ospf 1001
!
```

2. Configure a configuration group and apply it in a configuration submode:

```
RP/0/RP0/CPU0:router# show running group go-a

group go-a
 router ospf '.*'
  area '.*'
   interface 'Gig.*'
    cost 200
   !
  !
 !
end-group

RP/0/RP0/CPU0:router# show running group go-b

group go-b
 router ospf '.*'
  area '.*'
   interface 'Gig.*'
    cost 250
   !
  !
 !
end-group

RP/0/RP0/CPU0:router# show running group go-c

group go-c
 router ospf '.*'
  area '.*'
   interface 'Gig.*'
```

```
    cost 300
   !
  !
 !

end-group
```

3. Check the inheritance view and verify that the apply-group in the lowest configuration submode gets the highest priority:

```
RP/0/RP0/CPU0:router# show running inheritance router ospf 1

router ospf 1
 nsr
 router-id 121.121.121.121
 nsf cisco
 redistribute connected
 address-family ipv4 unicast
 area 0
  interface GigabitEthernet0/0/0/0
   ## Inherited from group go-a
   cost 200                    << apply-group in lowest submode  gets highest priority

  !
  interface GigabitEthernet0/0/0/1
   ## Inherited from group go-b
   cost 250
  !
  interface GigabitEthernet0/0/0/3
   ## Inherited from group go-b
   cost 250
  !
  interface GigabitEthernet0/0/0/4
   ## Inherited from group go-b
   cost 250
  !
  interface GigabitEthernet0/0/0/21
   bfd minimum-interval 100
   bfd fast-detect
   bfd multiplier 3
   ## Inherited from group go-b
   cost 250
  !
  interface TenGigE0/2/0/0.3891
  !
  interface TenGigE0/2/0/0.3892
  !
  interface TenGigE0/2/0/0.3893
  !
  interface TenGigE0/2/0/0.3894
  !
 !
!
```

4. Check the functionality of the cost inheritance through the groups:

```
RP/0/RP0/CPU0:router# show ospf 1 interface GigabitEthernet 0/0/0/0

GigabitEthernet0/0/0/0 is up, line protocol is up
  Internet Address 1.0.1.1/30, Area 0
  Process ID 1, Router ID 121.121.121.121, Network Type BROADCAST, Cost: 200
  Transmit Delay is 1 sec, State DR, Priority 1, MTU 1500, MaxPktSz 1500
  Designated Router (ID) 121.121.121.121, Interface address 1.0.1.1
```

```
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Non-Stop Forwarding (NSF) enabled
    Hello due in 00:00:02
  Index 5/5, flood queue length 0
  Next 0(0)/0(0)
  Last flood scan length is 1, maximum is 40
  Last flood scan time is 0 msec, maximum is 7 msec
  LS Ack List: current length 0, high water mark 0
  Neighbor Count is 1, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
  Multi-area interface Count is 0
```

Link Bundling Usage: Example

This example shows how to configure interface membership in a bundle link:

1. Configure the configuration groups:

```
RP/0/RP0/CPU0:router# show running group bundle1

group bundle1
 interface 'GigabitEthernet0/1/0/1[1-6]'
  bundle id 1 mode active
 !
end-group

RP/0/RP0/CPU0:router# show running | inc apply-group

Building configuration...

apply-group isis l2tr  isis2  mpp  bundle1
```

2. Check the local configuration:

```
RP/0/RP0/CPU0:router# show running interface gigabitEthernet 0/1/0/11

interface GigabitEthernet0/1/0/11
!

RP/0/RP0/CPU0:router# show running interface Bundle-Ether1

interface Bundle-Ether1
 ipv4 address 108.108.1.1 255.255.255.0
 bundle maximum-active links 10
 bundle minimum-active links 5
!
```

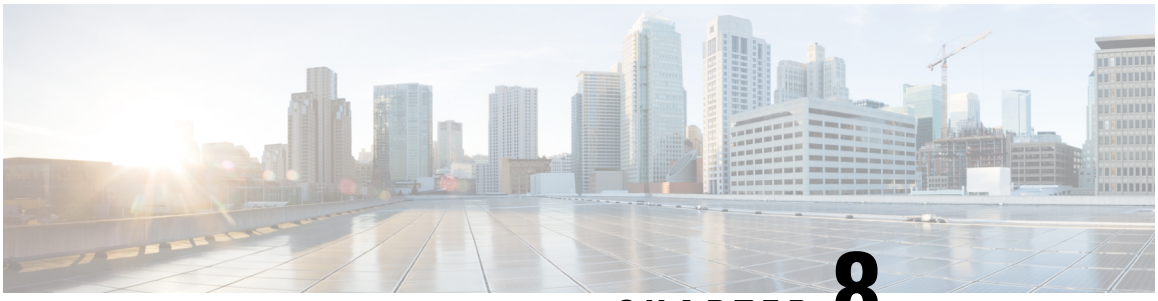
3. Check the inheritance configuration view:

```
RP/0/RP0/CPU0:router# show running inheritance interface GigabitEthernet 0/1/0/11

interface GigabitEthernet0/1/0/11
 ## Inherited from group bundle1
```

```
 bundle id 1 mode active
!
```

4. Check that the inheritance configuration took effect:

```
RP/0/RP0/CPU0:router# show interface Bundle-Ether1

Bundle-Ether1 is up, line protocol is up
  Interface state transitions: 1
  Hardware is Aggregated Ethernet interface(s), address is 0024.f71f.4bc3
  Internet address is 108.108.1.1/24
  MTU 1514 bytes, BW 6000000 Kbit (Max: 6000000 Kbit)
     reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Full-duplex, 6000Mb/s
  loopback not set,
  ARP type ARPA, ARP timeout 04:00:00
    No. of members in this bundle: 6
      GigabitEthernet0/1/0/11      Full-duplex  1000Mb/s     Active
      GigabitEthernet0/1/0/12      Full-duplex  1000Mb/s     Active
      GigabitEthernet0/1/0/13      Full-duplex  1000Mb/s     Active
      GigabitEthernet0/1/0/14      Full-duplex  1000Mb/s     Active
      GigabitEthernet0/1/0/15      Full-duplex  1000Mb/s     Active
      GigabitEthernet0/1/0/16      Full-duplex  1000Mb/s     Active
  Last input 00:00:00, output 00:00:00
  Last clearing of "show interface" counters never
  5 minute input rate 8000 bits/sec, 1 packets/sec
  5 minute output rate 3000 bits/sec, 1 packets/sec
     2058 packets input, 1999803 bytes, 426 total input drops
     0 drops for unrecognized upper-level protocol
     Received 1 broadcast packets, 2057 multicast packets
              0 runts, 0 giants, 0 throttles, 0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     1204 packets output, 717972 bytes, 0 total output drops
     Output 2 broadcast packets, 1202 multicast packets
     0 output errors, 0 underruns, 0 applique, 0 resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```



CHAPTER 8

Upgrading Field-Programmable Devices

In general terms, *field-programmable devices* (FPDs) are hardware devices implemented on router cards that support separate software upgrades. A *field-programmable gate array* (FPGA) is a type of programmable memory device that exists on most hardware components of the router. The term *FPD* has been introduced to collectively and generically describe any type of programmable hardware device on SIPs and shared port adapters (SPAs), including FPGAs. Cisco IOS XR software provides the Cisco FPD upgrade feature to manage the upgrade of FPD images on SIPs and SPAs.

This chapter describes the information that you must know to verify image versions and to perform an upgrade for SPA or SIP FPD images when incompatibilities arise.

For complete descriptions of the FPD commands listed in this module, refer to the upcoming sections. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 22: Feature History for Upgrading FPD Software on Cisco IOS XR Software

Release	Modification
Release 3.2	SIPs and SPAs were introduced. FPD images were introduced to support SIPs and SPAs.
Release 3.3.0	Reload option was added to the **upgrade hw-module fpd** command, including a prompt to inform the user. Possibility for multiple FPD images on a card was added.
Release 5.3.2	Enhance FPD upgrade and downgrade behavior.

This module contains the following topics:

Upgrading Field-Programmable Device

An FPD is a field programmable logic device which contains non-volatile, re-programmable memory to define its internal wiring and functionality. The contents of this non-volatile memory are called the FPD image or FPD firmware. Over the lifespan of an FPD, FPD firmware images may need upgrades for bug fixes or functionality improvements. These upgrades are performed in the field with minimum system impact.

Prerequisites for FPD Image Upgrades

You must install the FPD pie before you install the SMUs or Service Packs. If you install the SMU or Service Packs before the FPD pie, the FPDs on the line card may not upgrade. In such cases, you must remove the SMUs and Service Packs and reload the router.

Overview of FPD Image Upgrade Support

An FPD image is used to upgrade the software on an FPD.

Whenever an image is released that supports SIPs and SPAs, a companion SIP and SPA FPD image is bundled. Generally, the FPD image is not automatically upgraded. You must manually upgrade the FPD image running on the SPA or SIP when you upgrade the Cisco IOS XR software image.

FPD versions must be compatible with the Cisco IOS XR software that is running on the router; if an incompatibility exists between an FPD version and the Cisco IOS XR software, the device with the FPGA may not operate properly until the incompatibility is resolved. An FPGA incompatibility on a SPA does not necessarily affect the running of the SPA interfaces; an FPD incompatibility on a SIP disables all interfaces for all SPAs in the SIP until the incompatibility is addressed.

Use the **show hw-module fpd** command to determine if an FPD upgrade is required. A value of 'Yes' in the Upg/Dng? (upgrade/downgrade) column indicates that an upgrade or downgrade is required.

The NCS 5500 supports upgrades for FPGA devices on its SIPs and SPAs. FPGA and ROMMON software upgrades are part of an FPD image package that corresponds to a Cisco IOS XR software image. SIPs and SPAs support manual upgrades for FPGA devices using the Cisco FPD upgrade feature that is further described in this chapter.

Note It is mandatory to upgrade all the required FPDs before doing a reload when you are upgrading FPDs on line cards. This is because, partial FPD component upgrades might result in booting errors (in some cases).

Automatic FPD Upgrade

By default, the FPD image is not automatically upgraded. You must manually upgrade the FPD image running on the SPA or SIP when you upgrade the Cisco IOS XR software image.

However, if you enable the **fpd auto-upgrade** command in Admin Configuration mode, FPD images are automatically updated when:

- Software upgrade is carried out.

The following conditions must be met for an Automatic FPD Upgrade to work:

- The **fpd auto-upgrade** command must be enabled.

Note Although the FPD upgrade is performed during the install operation, there is no install commit performed. Therefore, once the FPD has been upgraded, if the image is rolled back to the original version, the FPD version is not downgraded to the previous version.

Automatic FPD Upgrade is not performed when:

- Line cards or other cards such as RPs, SPAs or alarm cards are added to an existing router.
- A line card chassis is added to an existing CRS multi-chassis router.
- A non-reload software maintenance upgrade (SMU) or PIE installation is performed, even where the FPD image version changes. Since a non-reload installation is, by definition, not supposed to reload the router, and an FPD upgrade requires a router reload, an Automatic FPD Upgrade is repressed.

Note In all cases where the automatic FPD upgrade is not performed, you must perform a manual FPD upgrade using the **upgrade hw-module fpd** command.

Note A message is displayed when router modules cannot get upgraded during automatic FPD upgrade indicating that the FPGA is intentionally skipped during upgrade. To upgrade such FPGAs, you can use the CLI command with a particular location explicitly specified. For example, **upgrade hw-module fpd all location 0/3/1**.

Note CFP2-DCO Optical modules do not support automatic-FPD upgrade.

FPD upgrade service

The main tasks of the FPD upgrade service are:

- FPD image version checking to decide if a specific firmware image needs an upgrade or not.
- Automatic FPD Image Upgrade (if enabled).
- Manual FPD Image Upgrade using the **upgrade hw-module fpd** command.
- Invoke the appropriate device driver with a name of the new image to load.

An FPD image package is used to upgrade FPD images. The **install activate** command is used to place the FPD binary files into the expected location on the boot devices.

Supported Upgrade Methods

Method	Remarks
Manual Upgrade	Upgrade using CLI, force upgrade supported.
Auto Upgrade	Upgrade using install SMU activation or during image upgrade. User can enable/disable auto upgrade feature.

Determining Upgrade Requirement

Use the **show hw-module fpd** command to determine if an FPD upgrade is required. Check for NEED UPGD in the Status column.

Use the **show fpd package** command to find out which FPGAs are supported with your current software release and minimum hardware requirements for each module.

Automatic FPD upgrade

Use the **fpd auto-upgrade enable** command to enable the auto upgrade feature.

The FPD images are upgraded as part of the install activation of the new image. The FPDs are upgraded before the router is reloaded.

During an FPD auto-upgrade, the installed FPD rpm package includes an FPD image with a new version of software that is different than the version of the image running on the hardware. Once the FPDs have been upgraded, even if the base image is rolled backed to the older version, the FPD will not be downgraded to its previous version.

When a reload package is installed with new FPD images, the FPD images are upgraded before the router gets reloaded. This feature is controlled through an fpd auto-upgrade configuration option. The auto-upgrade feature does not address the following:

- FPD Upgrade during initial boot
- FPD Upgrade during new card insertion

Manual FPD upgrade

Manual FPD upgrade is performed using the **upgrade hw-module fpd** command. All cards or all of FPGA in a card can be upgraded. If reload is required to activate FPD , the upgrade should be complete. All line-cards, fabric cards and RP cards cannot be reloaded during the process of the FPD upgrade.

FPD upgrade is transaction-based:

- Each fpd upgrade cli execution is one transaction
- Only one transaction is allowed at any given time
- One transaction may include one or many FPD upgrade(s)

The **force** option can be used to forcibly upgrade the FPD (regardless of whether it is required or not). It triggers all FPDs to be upgraded or downgraded. The force option can also be used to downgrade or upgrade the FPGAs even after the version check.

Note In some cases, FPDs can have primary and backup images.

How to Upgrade FPD Images

You must determine if an FPD image upgrade is needed using the **show hw-module fpd** command and perform the upgrade, if needed, under the following circumstances:

- You migrate the software to a later Cisco IOS XR software release.
- You swap SPAs or SIPs from a system running a different Cisco IOS XR software release.
- You insert a new SPA or SIP.

In the event that there is an FPD incompatibility with your card, you may receive an error message. If you upgrade to a newer version of the Cisco IOS XR software and there is an FPD incompatibility, you receive the following message:

```
LC/0/1/CPU0:Dec 23 16:33:47.945 : spa_192_jacket_v2[203]: %PLATFORM-UPGRADE_FPD-4-DOWN_REV
: spa fpga2 instance 0 is down-rev (V0.6), upgrade to (V1.0). Use the "upgrade hw-module
fpd" CLI in admin mode.
```

If the FPD image on the card is newer then what is required by the currently running Cisco IOS XR software image on the router, you receive the following error message:

```
LC/0/1/CPU0:Dec 23 16:33:47.955 : spa_192_jacket_v2[203]: %PLATFORM-UPGRADE_FPD-4-UP_REV :
spa fpga instance 1 is up-rev(V1.10), downgrade to (V0.09) is "OPTIONAL". Use "upgrade
hw-module fpd force" CLI in admin mode.
```

You should perform the FPD upgrade procedure if you receive such messages. Cards may not function properly if FPD incompatibilities are not resolved.

Note The use of the **force** option when performing a FPD upgrade is not recommended except under explicit direction from Cisco engineering or TAC.

Before you begin

- The FPD upgrade procedure is performed while the card is online. At the end of the procedure the card must be reloaded before the FPD upgrade is complete. To automatically reload the card, you can use the **hw-module reload** command during your next maintenance window. The upgrade procedure is not complete until the card is reloaded.
- During the FPD upgrade, you *must not* do the following:
  - Reload, perform an online insertion and removal (OIR) of a line card (LC), or power down the chassis. Doing so may cause the node to enter an unusable state.
  - Press **Ctrl-C** if the console appears to hang without any output. Doing so may abort the upgrade.
- If you are not sure whether a card requires an FPD upgrade, you can install the card and use the

show hw-module fpd command to determine if the FPD image on the card is compatible with the currently running Cisco IOS XR software release.

SUMMARY STEPS

1. **show hw-module fpd location** {**all** | *node-id*}
2. **admin**
3. (Optional) **show fpd package**
4. **upgrade hw-module fpd** {**all** | *fpga-type*} [**force**] **location** [**all** | *node-id*]
5. **exit**
6. (Optional) **hw-module** {**location** *node-id* | **subslot** *subslot-id*} **reload**
7. **show platform**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show hw-module fpd location** {**all** | *node-id*}
	Example:
	`RP/0/RP0/CPU0:router# show hw-module fpd location all`
	or
	`RP/0/RP0/CPU0:router# show hw-module fpd location 0/4/cpu0`
		Displays the current FPD image versions for the specified card or all cards installed in the router. Use this command to determine if you must upgrade the FPD image on your card.

Step 2	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.
Step 3	(Optional) **show fpd package** **Example:** `RP/0/RP0/CPU0:router(admin)# show fpd package`	Displays which cards are supported with your current Cisco IOS XR software release, which FPD image you need for each card, and what the minimum hardware requirements are for the various modules. (A minimum hardware requirement version of 0.0 indicates that all hardware can support this FPD image version.) If there are multiple FPD images for your card, use this command to determine which FPD image to use if you want to upgrade only a specific FPD type.
Step 4	**upgrade hw-module fpd** {**all** | *fpga-type*} [**force**] **location** [**all** | *node-id*] **Example:** `RP/0/RP0/CPU0:router(admin)# upgrade hw-module fpd all location 0/3/1`	Upgrades all the current FPD images that must be upgraded on the specified card with new images. Before continuing to the next step, wait for confirmation that the FPD upgrade has successfully completed. Status messages, similar to these, are displayed to the screen until the FPD upgrade is completed:

	Command or Action	Purpose
	. . . Successfully upgraded 1 FPD for SPA-2XOC48POS/RPR on location 0/3/1 RP/0/RP0/CPU0:V3_DC_MT(admin)# upgrade hw-module fpd all location all RP/0/RP0/CPU0:May 14 22:06:38.715 : upgrade_fpd_cli[65878]: %PLATFORM-UPGRADE_FPD-6-STATUS_UPG_LOC_ALL_OPT : pm fpga11 instance 14 on location 0/RP0/CPU0 was intentionally skipped during upgrade using location all option	FPD upgrade started. FPD upgrade in progress.. FPD upgrade in progress.. FPD upgrade sent to location xxxx FPD upgrade sent to location yyyy FPD upgrade in progress.. FPD upgrade finished for location xxx FPD upgrade in progress.. FPD upgrade finished for location yyyy FPD upgrade completed. The “FPD upgrade in progress.” message is printed every minute. These logs are information logs, and as such, are displayed if the **logging console informational** command is configured. If Ctrl-C is pressed while the FPD upgrade is in progress, the following warning message is displayed: FPD upgrade in progress on some hardware, aborting now is not recommended as it might cause HW programming failure and result in RMA of the hardware. Do you want to continue? [Confirm(y/n)] If you confirm that you want to abort the FPD upgrade procedure, this message is displayed: FPD upgrade process has been aborted, please check the status of the hardware and reissue the upgrade command if required. **Note** If your card supports multiple FPD images, you can use the **show fpd package** admin command to determine what specific image to upgrade in the **upgrade hw-module fpd** command. **Note** A message is displayed when router modules cannot get upgraded during upgrade with **location all** option indicating that the FPGA is intentionally skipped during upgrade. To upgrade such FPGAs, you can use the CLI command with a particular location explicitly specified. For example, **upgrade hw-module fpd all location 0/3/1**.
Step 5	**exit** **Example:** RP/0/RP0/CPU0:router(admin)# exit	Exits administration EXEC mode and returns to EXEC mode.

	Command or Action	Purpose

Step 6	(Optional) **hw-module** {**location** *node-id* | **subslot** *subslot-id*} **reload**
	Example:
	`RP/0/RP0/CPU0:router# hw-module subslot 0/3/1 reload`
	or
	`RP/0/RP0/CPU0:router# hw-module location 0/3/cpu0 reload`
		Use the **hw-module subslot reload** command to reload a SPA and the **hw-module location reload** command to reload a SIP or line card.

Step 7	**show platform** **Example:** `RP/0/RP0/CPU0:router# show platform`	Verifies that the FPD image on the card has been successfully upgraded by displaying the status of all cards in the system.

Configuration Examples for FPD Image Upgrade

The following examples indicates the use of commands associated with the FPD image upgrade procedure.

show hw-module fpd Command Output: Example

Use the **show hw-module fpd** to display the current version of FPD images on the SPAs, SIPs and other cards installed on your router.

This command can be used to identify information about FPDs on any card. If you enter the location of a line card that is not a SPA, the output displays information about any programmable devices on that line card.

The following example shows how to display FPD compatibility for all modules in the router:

```
RP/0/RSP0/CPU0:router# show hw-module fpd location all

==================================== ========================================
                                     Existing Field Programmable Devices
                                     ========================================
                                       HW                          Current SW Upg/
Location     Card Type                Version Type Subtype Inst   Version   Dng?
============ ======================== ======= ==== ======= ==== =========== ====
0/RSP0/CPU0  CRS1-SIP-800              1.0     lc   fpga3    0      1.23     Yes
                                                    fpga1    0      1.05      No
                                                    fpga2    0      3.08^     No
-------------------------------------------------------------------------------
0/0/0        SPA-2XCHOC12/DS0          1.0     spa  rommon   0      2.02      No
                                               spa   fpga     0      1.36+     No
                                               spa   fpga2    0      1.00*     No
-------------------------------------------------------------------------------
NOTES:
1.  One or more FPD needs an upgrade or a downgrade.  This can be accomplished
    using the "admin upgrade hw-module fpd" CLI.
```

```
2.  * One or more FPD is running minimum software version supported.
     It can be upgraded using the "admin> upgrade hw-module fpd <fpd> force location <loc>"
 CLI.
3.  + One or more FPD is running up-rev FPGA version. Downgrade is "OPTIONAL" in this case.

      It can be downgraded using the "admin> upgrade hw-module fpd <fpd> force location
<loc>" CLI.
4.  ^ One or more FPD will be intentionally skipped from upgrade using CLI with option "all"
 or during
     "Auto fpd". It can be upgraded only using the "admin> upgrade hw-module fpd <fpd>
location
     <loc>" CLI with exact location.
```

Note After Release 5.3.x, Upg/Dng? will display Yes only for upgrade.

The following example shows the FPD for which upgrage will be skipped.

```
RP/0/RP0/CPU0:router# show hw-module fpd location all

==================================== ========================================
                                     Existing Field Programmable Devices
                                     ========================================
                                       HW                        Current SW Upg/
Location     Card Type               Version Type Subtype Inst   Version    Dng?
============ ======================= ======= ==== ======= ==== =========== ====
0/SM1/SP     140G-4-S1S2S3             0.1  lc   rommonA   0    2.08        Yes
------------------------------------------------------------------------------
                                           lc    rommon    0    2.08        Yes
------------------------------------------------------------------------------
                                           lc    fpqa1     0    6.04^       No
------------------------------------------------------------------------------
                                           lc    fpga2     0    4.01        No
------------------------------------------------------------------------------
NOTES:
1. ^ One or more FPD will be intentionally skipped from upgrade using CLI with option "all"
 or during "Auto fpd".
    It can be upgraded only using the "admin> upgrade hw-module fpd <fpd> location <loc>"
CLI with exact location.

RP/0/RP0/CPU0:router# show hw-module fpd location 0/6/cpu0

Sun Apr 18 03:18:24.903 DST

==================================== ========================================
                                     Existing Field Programmable Devices
                                     ========================================
                                       HW                        Current SW Upg/
Location     Card Type               Version Type Subtype Inst   Version    Dng?
============ ======================= ======= ==== ======= ==== =========== ====
0/6/CPU0     CRS1-SIP-800              0.96  lc   fpga1   0       6.00     No
                                           lc   rommonA 0       2.100    No
                                           lc   rommon  0       2.100    No
------------------------------------------------------------------------------
```

If the cards in the system do not meet the minimum requirements, the output contains a “NOTES” section that states how to upgrade the FPD image.

Table 23: show hw-module fpd Field Descriptions

Field	Description
Location	Location of the module in the *rack*/*slot*/*module* notation.
Card Type	Module part number.
HW Version	Hardware model version for the module.
Type	Hardware type. Can be one of the following types: • spa—Shared port adapter • lc—Line card
Subtype	FPD type. Can be one of the following types: • fabldr—Fabric downloader • fpga1—Field-programmable gate array • fpga2—Field-programmable gate array 2 • fpga3—Field-programmable gate array 3 • fpga4—Field-programmable gate array 4 • fpga5—Field-programmable gate array 5 • rommonA—Read-only memory monitor A • rommon—Read-only memory monitor B
Inst	FPD instance. The FPD instance uniquely identifies an FPD and is used by the FPD process to register an FPD.
Current SW Version	Currently running FPD image version.
Upg/Dng?	Specifies whether an FPD upgrade or downgrade is required. A downgrade is required in rare cases when the version of the FPD image has a higher major revision than the version of the FPD image in the current Cisco IOS XR software package.

show fpd package Command Output: Example

Use the **show fpd package** command in administration EXECAdmin EXEC mode mode to find out which SPAs and SIPs are supported with your current Cisco IOS XR software release, which FPD image package you need for each SPA or SIP, and what the minimum hardware requirements are for each module. If multiple FPD images are available for your card, they are listed as Subtype fpga2, fpga3, and so on.

Note The FPD name used in the FPD Description column of the output of the `show fpd package` command includes the last ten characters of DCO-PID. Depending on the slot and port numbers, the FPD name is appended with DCO_0, DCO_1, or DCO_2. For example, the FPD names for `CFP2-WDM-D-1HL` in port 0 and port 1 are `-WDM-D-1HL_DCO_0` and `WDM-D-1HL_DCO_1` respectively.

The following example shows sample output from the **show fpd package** command:

```
show fpd package
Tue Jan 22 13:56:00.212 UTC

================================ ================================================
                                          Field Programmable Device Package
                                 ================================================
                                                  Req     SW       Min Req   Min Req
Card Type            FPD Description             Reload  Ver      SW Ver  Board Ver
================== ========================== ====== ======= ======== =========
------------------------------------------------------------------------------
NC55-1200W-ACFW      LIT-PriMCU-ACFW(A)          NO      2.09    2.09    0.0
------------------------------------------------------------------------------
NC55-900W-ACFW-I     LIT-PriMCU-ACFW-I(A)        NO      1.04    1.04    0.0
------------------------------------------------------------------------------
NC55-900W-DCFW-I     LIT-PriMCU-DCFW-I(A)        NO      2.260   2.260   0.0
------------------------------------------------------------------------------
NC55-930W-DCFW-C     LIT-PriMCU-DCFW-C(A)        NO      2.259   2.259   0.0
------------------------------------------------------------------------------
NC55-MPA-12T-S       MPAFPGA                     YES     0.27    0.27    0.0
------------------------------------------------------------------------------
NC55-MPA-1TH2H-S     -WDM-D-1HL_DCO_2            NO     38.518  38.518   0.1
                     MPAFPGA                     YES     0.53    0.53    0.0
                     WDM-DE-1HL_DCO_2            NO     38.518  38.518   0.1
                     WDM-DS-1HL_DCO_2            NO     38.268  38.268   0.1
------------------------------------------------------------------------------
NC55-MPA-2TH-HX-S    -WDM-D-1HL_DCO_0            NO     38.518  38.518   0.1
                     -WDM-D-1HL_DCO_1            NO     38.518  38.518   0.1
                     MPAFPGA                     YES     0.53    0.53    0.0
                     WDM-DE-1HL_DCO_0            NO     38.518  38.518   0.1
                     WDM-DE-1HL_DCO_1            NO     38.518  38.518   0.1
                     WDM-DS-1HL_DCO_0            NO     38.268  38.268   0.1
                     WDM-DS-1HL_DCO_1            NO     38.268  38.268   0.1
------------------------------------------------------------------------------
NC55-MPA-2TH-S       -WDM-D-1HL_DCO_0            NO     38.518  38.518   0.1
                     -WDM-D-1HL_DCO_1            NO     38.518  38.518   0.1
                     MPAFPGA                     YES     0.53    0.53    0.0
                     WDM-DE-1HL_DCO_0            NO     38.518  38.518   0.1
                     WDM-DE-1HL_DCO_1            NO     38.518  38.518   0.1
                     WDM-DS-1HL_DCO_0            NO     38.268  38.268   0.1
                     WDM-DS-1HL_DCO_1            NO     38.268  38.268   0.1
------------------------------------------------------------------------------
NC55-MPA-4H-HD-S     MPAFPGA                     YES     0.53    0.53    0.0
------------------------------------------------------------------------------
NC55-MPA-4H-HX-S     MPAFPGA                     YES     0.53    0.53    0.0
------------------------------------------------------------------------------
NC55-MPA-4H-S        MPAFPGA                     YES     0.53    0.53    0.0
------------------------------------------------------------------------------
NC55A2-MOD-SE-H-S    Bootloader(A)               YES     1.11    1.11    0.0
                     CPU-IOFPGA(A)               YES     1.18    1.18    0.1
                     MB-IOFPGA(A)                YES     0.18    0.18    0.1
                     MB-MIFPGA                   YES     0.19    0.19    0.0
                     SATA(A)                     NO      5.00    5.00    0.0
------------------------------------------------------------------------------
NCS-55A2-MOD-HD-S    Bootloader(A)               YES     1.11    1.11    0.0
                     CPU-IOFPGA(A)               YES     1.18    1.18    0.1
                     MB-IOFPGA(A)                YES     0.18    0.18    0.1
                     MB-MIFPGA                   YES     0.19    0.19    0.0
                     SATA(A)                     NO      5.00    5.00    0.0
------------------------------------------------------------------------------
NCS-55A2-MOD-HX-S    Bootloader(A)               YES     1.11    1.11    0.0
                     CPU-IOFPGA(A)               YES     1.18    1.18    0.1
                     MB-IOFPGA(A)                YES     0.18    0.18    0.1
                     MB-MIFPGA                   YES     0.19    0.19    0.0
                     SATA(A)                     NO      5.00    5.00    0.0
```

```
-------------------------------------------------------------------------------
NCS-55A2-MOD-S       Bootloader(A)              YES     1.11    1.11    0.0
                     CPU-IOFPGA(A)              YES     1.18    1.18    0.1
                     MB-IOFPGA(A)               YES     0.18    0.18    0.1
                     MB-MIFPGA                  YES     0.19    0.19    0.0
                     SATA(A)                    NO      5.00    5.00    0.0
-------------------------------------------------------------------------------
NCS-55A2-MOD-SE-S    Bootloader(A)              YES     1.11    1.11    0.0
                     CPU-IOFPGA(A)              YES     1.18    1.18    0.1
                     MB-IOFPGA(A)               YES     0.18    0.18    0.1
                     MB-MIFPGA                  YES     0.19    0.19    0.0
                     SATA(A)                    NO      5.00    5.00    0.0
                     STATSFPGA                  YES     0.01    0.01    0.0
```

This table describes the significant fields shown in the display:

Table 24: show fpd package Field Descriptions

Field	Description
Card Type	Module part number.
FPD Description	Description of all FPD images available for the SPA.
Type	Hardware type. Possible types can be: • spa—Shared port adapter • lc—Line card
Subtype	FPD subtype. These values are used in the **upgrade hw-module fpd** command to indicate a specific FPD image type to upgrade.
SW Version	FPD software version recommended for the associated module running the current Cisco IOS XR software.
Min Req SW Vers	Minimum required FPD image software version to operate the card. Version 0.0 indicates that a minimum required image was not programmed into the card.
Min Req HW Vers	Minimum required hardware version for the associated FPD image. A minimum hardware requirement of version 0.0 indicates that all hardware can support this FPD image version.

Note In the **show fpd package** command output, the "subtype" column shows the FPDs that correspond with each SPA image. To upgrade a specific FPD with the **upgrade hw-module fpd** command, replace the *fpga-type* argument with the appropriate FPD from the "subtype" column, as shown in the following example:

```
RP/0/RP0/CPU0:router(admin)# upgrade hw-module fpd fpga2 location 0/3/1 reload
```

upgrade hw-module fpd Command Output: Example

Use the **upgrade hw-module fpd** command to upgrade the FPD image on a SPA, SIP or line card.

```
RP/0/RP0/CPU0:Router# admin
RP/0/RP0/CPU0:Router(admin)# upgrade hw-module fpd fpga location 0/1/4

% RELOAD REMINDER:
  - The upgrade operation of the target module will not interrupt its normal
    operation. However, for the changes to take effect, the target module
    will need to be manually reloaded after the upgrade operation. This can
    be accomplished with the use of "hw-module <target> reload" command.
  - If automatic reload operation is desired after the upgrade, please use
    the "reload" option at the end of the upgrade command.
  - The output of "show hw-module fpd location" command will not display
    correct version information after the upgrade if the target module is
    not reloaded.
Continue? [confirm] y

SP/0/1/SP:Dec 22 05:41:17.920 : upgrade_daemon[125]: programming...with file /ne
t/node0_RP1_CPU0/hfr-lc-3.3.83/fpd/ucode/fpga_gladiator_sw0.6.xsvf
SP/0/1/SP:Dec 22 05:41:28.900 : upgrade_daemon[125]: ...programming...
SP/0/1/SP:Dec 22 05:41:28.906 : upgrade_daemon[125]: ...it will take a while...
SP/0/1/SP:Dec 22 05:41:29.004 : upgrade_daemon[125]: ...it will take a while...
SP/0/1/SP:Dec 22 05:43:03.432 : upgrade_daemon[125]: ...programming...
SP/0/1/SP:Dec 22 05:43:03.438 : upgrade_daemon[125]: ...it will take a while...
Successfully upgraded spa fpga instance 4 on location 0/1/4.
```

The following example shows how to upgrade the FPD image on a PLIM. Note that this upgrade does not upgrade the SPAs installed in the PLIM.

```
RP/0/RP0/CPU0:router(admin)# upgrade hw-module fpd all location 0/RP0/CPU0
Wed Mar 23 12:55:17.141 DST

***** UPGRADE WARNING MESSAGE: *****
  *  This upgrade operation has a maximum timout of 90 minutes.  *
  *  If you are executing the cmd for one specific location and  *
  *  card in that location reloads or goes down for some reason  *
  *  you can press CTRL-C to get back the RP's prompt.           *
  *  If you are executing the cmd for _all_ locations and a node *
  *  reloads or is down please allow other nodes to finish the   *
  *  upgrade process before pressing CTRL-C.                     *

% RELOAD REMINDER:
  - The upgrade operation of the target module will not interrupt its normal
    operation. However, for the changes to take effect, the target module
    will need to be manually reloaded after the upgrade operation. This can
    be accomplished with the use of "hw-module <target> reload" command.
  - If automatic reload operation is desired after the upgrade, please use
    the "reload" option at the end of the upgrade command.
  - The output of "show hw-module fpd location" command will not display
    correct version information after the upgrade if the target module is
    not reloaded.
NOTE: Chassis CLI will not be accessible while upgrade is in progress.
Continue? [confirm]

FPD upgrade in progress on some hardware, reload/configuration change
on those is not recommended as it might cause HW programming failure
and result in RMA of the hardware.

Starting the upgrade/download of following FPDs:
```

```
=========== ==== ======= ======= =========== =========
                                  Current    Upg/Dng
Location    Type Subtype Upg/Dng   Version    Version
=========== ==== ======= ======= =========== =========
0/6/CPU0    lc   fpga    upg         6.00        6.00
            lc   rommon  dng         2.100       2.03
-----------------------------------------------------
FPD upgrade in progress. Max timeout remaining 89 min.
FPD upgrade in progress. Max timeout remaining 88 min.
FPD upgrade in progress. Max timeout remaining 87 min.
Successfully upgraded fpga for CRS1-SIP-800 on location 0/6/CPU0 from 6.00 to 6.00
Successfully downgraded rommon for 40G-MSC on location 0/6/CPU0 from 2.100 to 2.03

FPD upgrade has ended.
```

show platform Command Output: Example

Use the **show platform** command to verify that the SPA is up and running.

```
RP/0/RP0/CPU0:router# show platform

Node            Type            PLIM            State           Config State
-------------------------------------------------------------------------------
0/1/SP          MSC(SP)         N/A             IOS XR RUN      PWR,NSHUT,MON
0/1/CPU0        MSC             Jacket Card     IOS XR RUN      PWR,NSHUT,MON
0/1/0           MSC(SPA)        4XOC3-POS       OK              PWR,NSHUT,MON
0/1/1           MSC(SPA)        OC192RPR-XFP    OK              PWR,NSHUT,MON
0/1/4           MSC(SPA)        8XOC3/OC12-POS  OK              PWR,NSHUT,MON
0/RP1/CPU0      RP(Active)      N/A             IOS XR RUN      PWR,NSHUT,MON
```

Troubleshooting Problems with FPD Image Upgrades

This section contains information to help troubleshoot problems that can occur during the upgrade process.

Power Failure or Removal of a SPA During an FPD Image Upgrade

If the FPD upgrade operation is interrupted by a power failure or the removal of the SPA, it could corrupt the FPD image. This corruption of the FPD image file makes the SPA unusable by the router and the system displays the following messages when it tries to power up the SPA. When it cannot successfully power up the SPA, it places it in the failed state, as shown in the following example:

```
LC/0/3/CPU0:Feb  4 08:23:16.672 : spa_192_jacket[188]: %L2-SPA-5-OIR_INSERTED : SPA discovered
 in bay 0
  LC/0/3/CPU0:Feb  4 08:23:23.349 : spa_192_jacket[188]: %L2-SPA-5-OIR_ERROR : SPA (0): An
 error occurred (0x1002), error recovery action: reset SPA
  LC/0/3/CPU0:Feb  4 08:23:26.431 : spa_192_jacket[188]: %L2-SPA-5-OIR_INSERTED : SPA
discovered in bay 0
  LC/0/3/CPU0:Feb  4 08:23:32.593 : spa_192_jacket[188]: %L2-SPA-5-OIR_ERROR : SPA (0): Too
 many retries, error recovery stopped
  LC/0/3/CPU0:Feb  4 08:23:32.593 : spa_192_jacket[188]: %L2-SPA-5-OIR_ERROR : SPA (0): An
 error occurred (0x1002), error recovery action: hold SPA in reset
```

When a SPA is in the failed state, it may not register itself with the FPD upgrade mechanism. In this case, you do not see the SPA listed when you use the **show hw-module fpd** command. To verify the state of a SPA, use the **show hw-module subslot error** command and the **show hw-module subslot status** command.

Performing a SPA FPD Recovery Upgrade

To recover a SPA from the failed state because of a corrupted FPD image, you must manually shut down the SPA. Use the **hw-module subslot** *subslot-id* **shutdown** command in Global Configuration mode to administratively shutdown the SPA. After the SPA is shut down, you can use the **upgrade hw-module fpd** command in administration EXEC mode:

```
RP/0/RP0/CPU0:router# admin
RP/0/RP0/CPU0:router(admin)# upgrade hw-module fpd fpga location 0/3/0
```

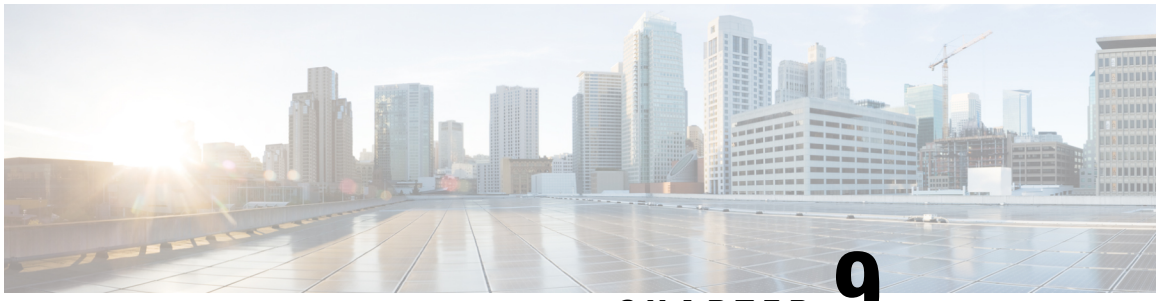
Performing a SIP FPD Recovery Upgrade

If a SIP upgrade fails for whatever reason, do not reload the SIP. Try to perform the upgrade procedure again. You can perform the upgrade procedure multiple times, as long as you do not reload the SIP. The FPD upgrade procedure takes several minutes to complete; do not interrupt the procedure. If you reload the SIP when the FPD image is corrupted, the SIP malfunctions and you must contact Cisco technical support for assistance.

To recover a SIP from the failed state because of a corrupted FPD image, you must contact Cisco technical support.

To recover a SIP from the failed state because of a corrupted FPD image, you must turn off the automatic reset of the SIP card. Use the **hw-module reset auto disable** command in administration configuration mode, as shown in the following example:

```
RP/0/RP0/CPU0:router(admin-config)# hw-module reset auto disable location 0/1/4
```



CHAPTER 9

Configuring Manageability

This module describes the configuration required to enable the Extensible Markup Language (XML) agent services. The XML Parser Infrastructure provides parsing and generation of XML documents with Document Object Model (DOM), Simple Application Programming Interface (API) for XML (SAX), and Document Type Definition (DTD) validation capabilities:

- DOM allows customers to programmatically create, manipulate, and generate XML documents.
- SAX supports user-defined functions for XML tags.
- DTD allows for validation of defined document types.

Table 25: Feature History for Configuring Manageability on Cisco IOS XR Software

Release 3.2	This feature was introduced.
Release 3.8.0	An enhanced-performance XML agent was introduced.
Release 3.9.0	The ability to enable XML requests over Secure Socket Layer (SSL) was introduced. The ability to configure an idle timeout for the XML agent was introduced.
Release 4.0.0	The ability to configure a dedicated agent to receive and send messages via a specified VPN routing and forwarding (VRF) instance was introduced. The ability to control CPU time used by the XML agent was introduced.

This module contains the following topics:

Information About XML Manageability

The Cisco IOS XR Extensible Markup Language (XML) API provides a programmable interface to the router for use by external management applications. This interface provides a mechanism for router configuration and monitoring utilizing XML formatted request and response streams. The XML interface is built on top of

the Management Data API (MDA), which provides a mechanism for Cisco IOS XR components to publish their data models through MDA schema definition files.

Cisco IOS XR software provides the ability to access the router via XML using a dedicated TCP connection, Secure Socket Layer (SSL), or a specific VPN routing and forwarding (VRF) instance.

How to Configure Manageability

Configuring the XML Agent

SUMMARY STEPS

1. **xml agent** [**ssl**]
2. **iteration on size** *iteration-size*
3. **session timeout** *timeout*
4. **throttle** {**memory** *size* | **process-rate** *tags*}
5. **vrf** { **default** | *vrf-name*} [**access-list** *access-list-name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**xml agent** [**ssl**] **Example:** `RP/0/RP0/CPU0:router:router(config)# xml agent`	Enables Extensible Markup Language (XML) requests over a dedicated TCP connection and enters XML agent configuration mode. Use the **ssl** keyword to enable XML requests over Secure Socket Layer (SSL).
Step 2	**iteration on size** *iteration-size* **Example:** `RP/0/RP0/CPU0:router:router(config-xml-agent)# iteration on size 500`	Configures the iteration size for large XML agent responses in KBytes. The default is 48.
Step 3	**session timeout** *timeout* **Example:** `RP/0/RP0/CPU0:router:router(config-xml-agent)# session timeout 5`	Configures an idle timeout for the XML agent in minutes. By default, there is no timeout.
Step 4	**throttle** {**memory** *size*	**process-rate** *tags*} **Example:** `RP/0/RP0/CPU0:router:router(config-xml-agent)# throttle memory 300`	Configures the XML agent processing capabilities. • Specify the throttle memory size in Mbytes per session. Values can range from 100 to 600. The default is 300. • Specify the process-rate as the number of tags that the XML agent can process per second. Values can range from 1000 to 30000. By default the process rate is not throttled.

	Command or Action	Purpose
Step 5	**vrf** { **default** | *vrf-name* } [**access-list** *access-list-name*] **Example:** `RP/0/RP0/CPU0:router:router(config-xml-agent)# vrf my-vrf`	Configures the dedicated agent or SSL agent to receive and send messages via the specified VPN routing and forwarding (VRF) instance.

Configuration Examples for Manageability

Enabling VRF on an XML Agent: Examples

The following example illustrates how to configure the dedicated XML agent to receive and send messages via VRF1, VRF2 and the default VRF:

```
RP/0/RP0/CPU0:router:router(config)# xml agent
RP/0/RP0/CPU0:router:router(config-xml-agent)# vrf VRF1
RP/0/RP0/CPU0:router:router(config-xml-agent)# vrf VRF2
```

The following example illustrates how to remove access to VRF2 from the dedicated agent:

```
RP/0/RP0/CPU0:router:router(config)# xml agent
RP/0/RP0/CPU0:router:router(config-xml-agent)# no vrf VRF2
```

The following example shows how to configure the XML SSL agent to receive and send messages through VRF1, VRF2 and the default VRF:

```
RP/0/RP0/CPU0:router:router(config)# xml agent ssl
RP/0/RP0/CPU0:router:router(config-xml-agent)# vrf VRF1
RP/0/RP0/CPU0:router:router(config-xml-agent)# vrf VRF2
```

The following example removes access for VRF2 from the dedicated XML agent:

```
RP/0/RP0/CPU0:router:router(config)# xml agent ssl
RP/0/RP0/CPU0:router:router(config-xml-agent)# no vrf VRF2
```

Additional References

The following sections provide references related to configuring manageability on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Cisco IOS XR commands	*Cisco IOS XR Commands Master List for the Cisco CRS Router*
Cisco IOS XR XML API material	*Cisco IOS XR XML API Guide for the Cisco CRS Router*
Information about user groups and task IDs	*Configuring AAA Services on Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*

Standards and RFCs

Standard/RFC	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

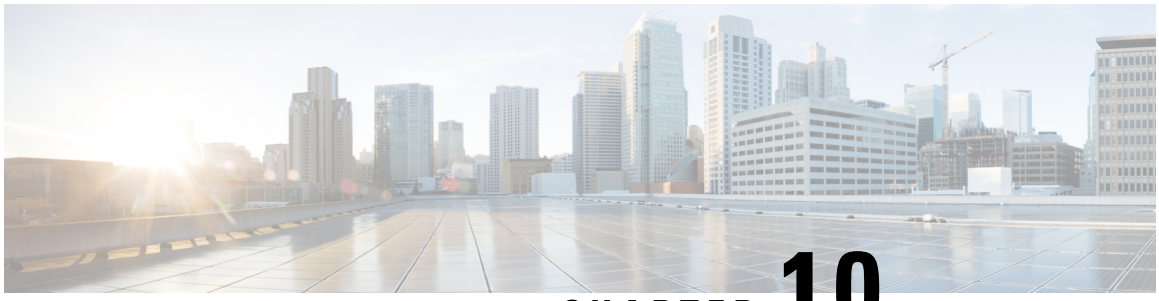
MIB	MIBs Link
—	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

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 10

Configuring Call Home

This module describes the configuring of the Call Home feature.

Table 26: Feature History for Configuring Call Home

Release	Modification
Release 4.1.0	Call Home was introduced

This model contains the following topics:

About Call Home

Call Home provides an email and http/https based notification for critical system policies. A range of message formats are available for compatibility with pager services or XML-based automated parsing applications. You can use this feature to page a network support engineer, email a Network Operations Center, or use Cisco

Smart Call Home services to generate a case with the Technical Assistance Center. The Call Home feature can deliver alert messages containing information about diagnostics and environmental faults and events.

The Call Home feature can deliver alerts to multiple recipients, referred to as Call Home destination profiles. Each profile includes configurable message formats and content categories. A predefined destination is provided for sending alerts to the Cisco TAC, but you also can define your own destination profiles. When you configure Call Home to send messages, the appropriate CLI show command is executed and the command output is attached to the message. Call Home messages are delivered in the following formats:

- Short text format which provides a one or two line description of the fault that is suitable for pagers or printed reports.
- Full text format which provides fully formatted message with detailed information that is suitable for human reading.
- XML machine readable format that uses Extensible Markup Language (XML) and Adaptive Messaging Language (AML) XML schema definition (XSD). The AML XSD is published on the Cisco.com website at http://www.cisco.com/. The XML format enables communication with the Cisco Systems Technical Assistance Center.

Destination Profiles

A destination profile includes the following information:

- One or more alert groups—The group of alerts that trigger a specific Call Home message if the alert occurs.
- One or more e-mail or http destinations—The list of recipients for the Call Home messages generated by alert groups assigned to this destination profile.
- Message format—The format for the Call Home message (short text, full text, or XML).
- Message severity level—The Call Home severity level that the alert must meet before a Call Home message is sent to all e-mail and http url addresses in the destination profile. An alert is not generated if the Call Home severity level of the alert is lower than the message severity level set for the destination profile.

You can also configure a destination profile to allow periodic inventory update messages by using the inventory alert group that will send out periodic messages daily, weekly, or monthly.

The following predefined destination profiles are supported:

- CiscoTAC-1—Supports the Cisco-TAC alert group in XML message format.

Call Home Alert Groups

An alert group is a predefined subset of alerts or events that Call Home detects and reports to one or more destinations. Alert groups allow you to select the set of alerts that you want to send to a predefined or custom destination profile. Alerts are sent to e-mail destinations in a destination profile only if that alert belongs to one of the alert groups associated with that destination profile and if the alert has a Call Home message severity at or above the message severity set in the destination profile.

The following table lists supported alert groups and the default CLI command output included in Call Home messages generated for the alert group.

Table 27: Alert Groups and Executed Commands

Alert Group	Description	Executed Commands
Environmental	Events related to power, fan, and environment-sensing elements such as temperature alarms.	**show environment** **show logging** **show inventory** **show environment trace** **show diag**
Inventory	Inventory status that is provided whenever a unit is cold booted, or when FRUs are inserted or removed. This alert is considered a noncritical event, and the information is used for status and entitlement.	Full-inventory messages: • **admin show diag** • **admin show diag chassis eeprom** • **admin show install active summary** • **admin show inventory** • **admin show sdr summary** • **admin show version** Delta OIR messages: • **admin show diag** [*location*] **details** • **admin show version** • **admin show inventory location** [*location*] • **admin show inventory active summary** • **admin show sdr summary**
Syslog	Events generated by specific interesting syslog messages	**admin show version** **admin show logging** **admin show inventory**
Configuration	User-generated request for configuration or configuration change event.	• **show version** • **show running config all** • **show inventory** • **show configuration history last 30** • **show configuration commit changes last 1**

Alert Group	Description	Executed Commands
Snapshot	This alert group can be configured for periodic notifications	By default, this alert group has no commands to be run. You can add the required commands that need to be run.

Call Home maps the syslog severity level to the corresponding Call Home severity level for syslog port group messages.

Call Home Message Levels

Call Home allows you to filter messages based on their level of urgency. You can associate each destination profile (predefined and user-defined) with a Call Home message level threshold. The Call Home message level ranges from 0 (lowest level of urgency) to 9 (highest level of urgency). Call Home messages are generated if they have a severity level equal to or greater than the Call Home message level threshold for the destination profile.

Call Home messages that are sent for syslog alert groups have the syslog severity level mapped to the Call Home message level.

Note Call Home does not change the syslog message level in the message text.

The following table lists each Call Home message level keyword and the corresponding syslog level for the syslog port alert group.

Table 28: Severity and syslog Level Mapping

Call Home Level	Keyword	syslog Level	Description
9	Catastrophic	N/A	Network-wide catastrophic failure.
8	Disaster	N/A	Significant network impact.
7	Fatal	Emergency (0)	System is unusable.
6	Critical	Alert (1)	Critical conditions that indicate that immediate attention is 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.

Obtaining Smart Call Home

If you have a service contract directly with Cisco Systems, you can register your devices for the Smart Call Home service. Smart Call Home provides fast resolution of system problems by analyzing Call Home messages sent from your devices and providing background information and recommendations. For issues that can be identified as known, particularly GOLD diagnostics failures, Automatic Service Requests will be generated with the Cisco-TAC.

Smart Call Home offers the following features:

- Continuous device health monitoring and real-time diagnostic alerts.
- Analysis of Call Home messages from your device and, where appropriate, Automatic Service Request generation, routed to the appropriate TAC team, including detailed diagnostic information to speed problem resolution.
- Secure message transport directly from your device or through a downloadable Transport Gateway (TG) aggregation point. You can use a TG aggregation point in cases that require support for multiple devices or in cases where security requirements mandate that your devices may not be connected directly to the Internet.
- Web-based access to Call Home messages and recommendations, inventory and configuration information for all Call Home devices. Provides access to associated field notices, security advisories and end-of-life information.

You need the following items to register:

- The SMARTnet contract number for your device
- Your e-mail address
- Your Cisco.com ID

For more information about Smart Call Home, see the Smart Call Home page at this URL:
https://supportforums.cisco.com/community/netpro/solutions/smart_services/smartcallhome

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. 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 is sent.

When Call Home is configured for anonymous reporting, only , inventory, and test messages are sent to Cisco. No identifying information is 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

Configuring Call Home

The tasks in this module describe how to configure the sending of Call Home messages. The following steps are involved:

1. Assign contact information.
2. Configure and enable one or more destination profiles.
3. Associate one or more alert groups to each profile.
4. Configure the email server options.
5. Enable Call Home.

Note Before enabling Call-Home, you must configure the source interface for http over IPv6. However, for http over IPv4, Call-Home works without the source interface.

Use the **http client source-interface ipv6** command to configure the source interface.

Configuring Contact Information

Each router must include a contact e-mail address. You can optionally include other identifying information for your system installation.

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **contact-email-addr** *email-address*
4. (Optional) **contract-id** *contract-id-string*
5. (Optional) **customer-id** *customer-id-string*
6. (Optional) **phone-number** *phone-number-string*
7. (Optional) **street-address** *street-address*
8. (Optional) **site-id** *site-id-string*
9. **commit**
10. **show call-home**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home** **Example:**	Enters call home configuration mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config)# call-home` `RP/0/RP0/CPU0:router(config-call-home)#`	
Step 3	**contact-email-addr** *email-address* **Example:** `RP/0/RP0/CPU0:router(config-call-home)# contact-email-addr user1@cisco.com`	Configures the customer email address. Enter up to 200 characters in email address format with no spaces.
Step 4	(Optional) **contract-id** *contract-id-string* **Example:** `RP/0/RP0/CPU0:router(config-call-home)# contract-id Contract-identifier`	Configures the contract ID. Enter up to 64 characters. If you include spaces, you must enclose the entry in quotes ("").
Step 5	(Optional) **customer-id** *customer-id-string* **Example:** `RP/0/RP0/CPU0:router(config-call-home)# customer-id Customer1`	Configures the customer ID. Enter up to 64 characters. If you include spaces, you must enclose the entry in quotes ("").
Step 6	(Optional) **phone-number** *phone-number-string* **Example:** `RP/0/RP0/CPU0:router(config-call-home)# phone-number +405-123-4567`	Configures the customer phone number. The number must begin with a plus (+) prefix, and may contain only dashes (-) and numbers. Enter up to 16 characters.
Step 7	(Optional) **street-address** *street-address* **Example:** `RP/0/RP0/CPU0:router(config-call-home)# street-address "300 E. Tasman Dr. San Jose, CA 95134"`	Configures the customer street address where RMA equipment can be shipped. Enter up to 200 characters. If you include spaces, you must enclose the entry in quotes ("").
Step 8	(Optional) **site-id** *site-id-string* **Example:** `RP/0/RP0/CPU0:router(config-call-home)# site-id SJ-RouterRoom1`	Configures the site ID for the system. Enter up to 200 characters. If you include spaces, you must enclose the entry in quotes ("").
Step 9	**commit**	
Step 10	**show call-home** **Example:** `RP/0/RP0/CPU0:router# show call-home`	Displays information about the system contacts.

Configuring and Activating Destination Profiles

You must have at least one activated destination profile for Call Home messages to be sent. The CiscoTAC-1 profile exists by default but is not active.

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **profile** *profile-name*
4. **destination address email** *email-address*
5. **destination message-size-limit** *max-size*
6. **destination preferred-msg-format** {**short-text** | **long-text** | **xml**}
7. **destination transport-method** [**email** | **hhtp**]
8. **active**
9. **commit**
10. **show call-home profile** {**all** | *profile-name*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

Step 2	**call-home**
	Example:
	`RP/0/RP0/CPU0:router(config)# call-home`
	`RP/0/RP0/CPU0:router(config-call-home)#`
		Enters call home configuration mode.

Step 3	**profile** *profile-name*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home)# profile my_profile`
	`RP/0/RP0/CPU0:router(config-call-home-profile)#`
		Enters call home profile configuration mode to configure a new or existing profile.

Step 4	**destination address email** *email-address*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home-profile)# destination`
	`address email support_me@cisco.com`
		Configures an email address to which Call Home messages are sent for this profile.

Step 5	**destination message-size-limit** *max-size*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home-profile)# destination`
	`message-size-limit 1000`
		Configures the maximum size of Call Home messages for this profile. Values can be between 50 and 3145728 characters.

	Command or Action	Purpose
Step 6	**destination preferred-msg-format** {**short-text** | **long-text** | **xml**} **Example:** `RP/0/RP0/CPU0:router(config-call-home-profile)# destination preferred-msg-format xml`	Configures the message format for this profile. The default is xml.
Step 7	**destination transport-method** [**email** | **hhtp**] **Example:** `RP/0/RP0/CPU0:router(config-call-home-profile)# destination transport-method email`	Configures the transport method for this profile.
Step 8	**active** **Example:** `RP/0/RP0/CPU0:router(config-call-home-profile)# active`	Activates the destination profile. **Note** At least one destination profile must be active for Call Home messages to be sent.
Step 9	**commit**	
Step 10	**show call-home profile** {**all** | *profile-name*} **Example:** `RP/0/RP0/CPU0:router# show call-home profile all`	Displays information about the destination profile.

Associating an Alert Group with a Destination Profile

An alert is sent only to destination profiles that have subscribed to the Call Home alert group.

Before you begin

Use the **show call-home alert-group** command to view available alert groups.

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **profile** *profile-name*
4. **subscribe-to-alert-group inventory** [**periodic** {**daily** | **monthly** *day-of-month* | **weekly** *day-of-week*} *hh:mm*
5. **subscribe-to-alert-group syslog severity** *severity-level* **pattern** *string*
6. **subscribe-to-alert-group snapshot severity** *severity-level* **pattern** *string*
7. **subscribe-to-alert-group configuration severity** *severity-level* **pattern** *string*
8. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

Step 2	**call-home**
	Example:
	`RP/0/RP0/CPU0:router(config)# call-home`
	`RP/0/RP0/CPU0:router(config-call-home)#`
		Enters call home configuration mode.

Step 3	**profile** *profile-name*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home)# profile my_profile`
	`RP/0/RP0/CPU0:router(config-call-home-profile)#`
		Enters call home profile configuration mode to configure a new or existing profile.

Step 4	**subscribe-to-alert-group inventory** [**periodic** {**daily** | **monthly** *day-of-month* | **weekly** *day-of-week*} *hh:mm*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group`
	`inventory periodic monthly 1 10:00`
		Configures a destination profile to receive messages for the inventory alert group. Either alerts are sent periodically, or any non-normal event triggers an alert.

Step 5	**subscribe-to-alert-group syslog severity** *severity-level* **pattern** *string*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group`
	`syslog severity major pattern`
		Configures a destination profile to receive messages for the syslog alert group. Alerts with a severity the same or greater than the specified severity level are sent.
		• **catastrophic**—Includes network-wide catastrophic events in the alert. This is the highest severity.
		• **critical**—Includes events requiring immediate attention (system log level 1).
		• **disaster**—Includes events with significant network impact.
		• **fatal**—Includes events where the system is unusable (system log level 0).
		• **major**—Includes events classified as major conditions (system log level 2).
		• **minor**—Includes events classified as minor conditions (system log level 3)
		• **normal**—Specifies the normal state and includes events classified as informational (system log level 6). This is the default.
		• **notification**—Includes events informational message events (system log level 5).

	Command or Action	Purpose
		• **warning**—Includes events classified as warning conditions (system log level 4). You can specify a pattern to be matched in the syslog message. If the pattern contains spaces, you must enclose it in quotes ("").
Step 6	**subscribe-to-alert-group snapshot severity** *severity-level* **pattern** *string* **Example:** `RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group snapshot severity major pattern`	Configures a destination profile to receive messages for the snapshot alert group. Alerts with a severity the same or greater than the specified severity level are sent. You can specify a pattern to be matched in the syslog message. If the pattern contains spaces, you must enclose it in quotes ("").
Step 7	**subscribe-to-alert-group configuration severity** *severity-level* **pattern** *string* **Example:** `RP/0/RP0/CPU0:router(config-call-home-profile)# subscribe-to-alert-group configuration severity major pattern`	Configures a destination profile to receive messages for the configuration alert group. Alerts with a severity the same or greater than the specified severity level are sent. You can specify a pattern to be matched in the syslog message. If the pattern contains spaces, you must enclose it in quotes ("").
Step 8	**commit**	

What to do next

Use the **show call-home profile** command to view the profile configurations.

Configuring Email

Call Home messages are sent via email. You must configure your email server before Call Home messages can be sent.

SUMMARY STEPS

1. **configure**
2. **call-home**
3. (Optional) **sender from** *email-address*
4. (Optional) **sender reply-to** *email-address*
5. **mail-server** *address* **priority** *priority*
6. **rate-limit** *events-count*
7. **commit**
8. **show call-home mail-server status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

Step 2	**call-home**
	Example:
	`RP/0/RP0/CPU0:router(config)# call-home`
	`RP/0/RP0/CPU0:router(config-call-home)#`
		Enters call home configuration mode.

Step 3	(Optional) **sender from** *email-address*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home)# sender from`
	`my_email@cisco.com`
		Specifies the email message "from" address.

Step 4	(Optional) **sender reply-to** *email-address*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home)# sender reply-to`
	`my_email@cisco.com`
		Specifies the email message "reply-to" address.

Step 5	Required: **mail-server** *address* **priority** *priority*
	Example:
	`RP/0/RP0/CPU0:router(config-call-home)# mail-server`
	`198.51.100.10 priority 1`
		Specifies the mail server to use to send Call Home messages. You can specify an IP address or mail server name. You can specify up to five mail servers to use. The server with the lower priority is tried first.

Step 6	Required: **rate-limit** *events-count* **Example:** `RP/0/RP0/CPU0:router(config-call-home)# rate-limit 4`	Specifies the maximum trigger rate per minute. The default is five events per minute and the maximum is also five.
Step 7	**commit**	
Step 8	**show call-home mail-server status** **Example:** `RP/0/RP0/CPU0:router# show call-home mail-server status`	Displays the status of the specified mail server.

Enabling Call Home

By default the sending of Call Home messages is disabled. You must perform this task to enable the sending of Call Home messages.

Before you begin

Before enabling the sending of Call Home messages, you should complete the configuration tasks described in this module. Specifically, you must have enabled a destination profile for any Call Home messages to be sent.

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **service active**
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

Step 2	**call-home**
	Example:
	`RP/0/RP0/CPU0:router(config)# call-home`
	`RP/0/RP0/CPU0:router(config-call-home)#`
		Enters call home configuration mode.

Step 3	**service active** **Example:** `RP/0/RP0/CPU0:router(config-call-home)# service active`	Enables the sending of Call Home messages.
Step 4	**commit**	

Configuring Smart Call Home (single command)

SUMMARY STEPS

1. **configure**
2. **call-home reporting** { **anonymous** | **contact-email** *email-address* } [**http-proxy** { *address* } **port** *port-number*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home reporting** { **anonymous**	**contact-email** *email-address* } [**http-proxy** { *address* } **port** *port-number*]	Enables all call home basic configurations using a single command.

	Command or Action	Purpose
	Example: `RP/0/RP0/CPU0:router (config) # call-home reporting contact-email email@company.com`	

Configuring Call Home Data Privacy

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **data-privacy** { **level** { **normal** | **high** } | **hostname** }

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home** **Example:** `RP/0/RP0/CPU0:router(config) # call-home`	Enters the call home configuration submode.
Step 3	**data-privacy** { **level** { **normal**	**high** }	**hostname** } **Example:** `RP/0/RP0/CPU0:router(config-call-home) # data-privacy level high`	Scrubs data from call-home message to protect the privacy of the user. The default data-privacy level is normal. • **normal** - scrubs all normal level commands , such as , password/ username/ ip/ destination. • **high** - scrubs all normal level commands plus the IP domain name and IP address commands. • **hostname** - scrubbing the hostname from call-home messages may cause Smart Call Home processing failure. **Note** Enabling the data-privacy command can affect CPU utilization when scrubbing a large amount of data.

Configuring Syslog Throttling

This task is used to enable or disable Call Home syslog message throttling and avoid sending repetitive Call Home syslog messages.

SUMMARY STEPS

1. **configure**

2. **call-home**
3. **syslog-throttling**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home** **Example:** `RP/0/RP0/CPU0:router (config) # call-home`	Enters call home configuration submode.
Step 3	**syslog-throttling** **Example:** `RP/0/RP0/CPU0:router (config-call-home) # syslog-throttling`	Enables or disables Call Home syslog message throttling and avoids sending repetitive Call Home syslog messages. By default, syslog message throttling is enabled.

Enabling AAA Authorization

This task is used to enable AAA authorization for Call Home messages.

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **aaa-authorization** [**username** *username*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home** **Example:** `RP/0/RP0/CPU0:router (config) # call-home`	Enters Call Home configuration mode.
Step 3	**aaa-authorization** [**username** *username*] **Example:** `RP/0/RP0/CPU0:router (config-call-home) # aaa-authorization username u1`	Enables AAA authorization. Specifies the username for authorization.

Sending Call Home Alert group Messages Manually

This task is used to manually trigger Call Home alert group messages.

You can use the **call-home send** command to manually send a specific alert group message. Guidelines for the CLI options of the command:

- Only the snapshot, configuration, and inventory alert groups can be sent manually. Syslog alert groups cannot be sent manually.
- When you manually trigger a 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 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.

SUMMARY STEPS

1. **call-home send alert-group snapshot** [**profile** *name*]
2. **call-home send alert-group configuration** [**profile** *name*]
3. **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:** `RP/0/RP0/CPU0:router # call-home send alert-group snapshot profile p1`	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 configuration** [**profile** *name*] **Example:** `RP/0/RP0/CPU0:router # call-home send alert-group configuration profile p1`	Sends a configuration alert group message to one destination profile if specified or to all subscribed destination profiles.
Step 3	**call-home send alert-group inventory** [**profile** *name*] **Example:** `RP/0/RP0/CPU0:router # call-home send alert-group inventory profile p1`	Sends an inventory alert group message to one destination profile if specified or to all subscribed destination profiles.

Manually sending command output message for a Command List

You can use the **call-home send** command to execute a command or a list of commands and send the command output through HTTP or email protocol.

Guidelines when sending the output of a command:

- The specified command or list of commands can be any run command, including commands for all modules. The command must be contained in quotes (“”).
- If the email option is selected using the “email” keyword and an email address is specified, the command output is sent to that address.
- If neither the email 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 email.
- If the HTTP option is specified, the CiscoTAC-1 profile destination HTTP or HTTPS URL is used as the destination. The destination email address can be specified so that Smart Call Home can forward the message to the email address. The user must specify either the destination email address or an SR number but they can also specify both.

This task enables you to execute command and send the command output.

SUMMARY STEPS

1. **call-home send** { *cli command* | *cli list* } [**email** *email* **msg-format** { **long-text** | **xml** } | **http** { **destination-email-address** *email* }] [**tac-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-request** *SR#*] **Example:** `RP/0/RP0/CPU0: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 email or HTTP. • { cli command | cli list }—Specifies the command or list of commands (separated by ‘;’). It can be any run command, including commands for all modules. The commands must be contained in quotes (“”). • **emai**l email **msg-format { long-text | xml }**—If the email option is selected, the command output will be sent to the specified email address in long-text or XML format with the service request number in the subject. The email address, the service request number, or both must be specified. The service request number is required if the email 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 Smart Call Home backend server (URL specified in the CiscoTAC-1 profile) in XML format. destination-email-address email can be specified so that the backend server can forward the message to the email address. The email address, the service request number, or both must be specified.

	Command or Action	Purpose
		• **tac-service-request** SR# —Specifies the service request number. The service request number is required if the email address is not specified.

Configuring a HTTP Proxy Server

This task enables the user to configure a HTTP Proxy Server.

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **http-proxy** *proxy-server-name* **port** *port-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home** **Example:** `RP/0/RP0/CPU0:router (config) # call-home`	Enters Call Home configuration mode.
Step 3	**http-proxy** *proxy-server-name* **port** *port-number* **Example:** `RP/0/RP0/CPU0:router (config) # http-proxy p1 port 100`	Configures the port for the specified HTTP proxy server. Range is 1 to 65535.

Configuring Snapshot alert group

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **alert-group-configuration snapshot**
4. **add-command** *"command string"*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

	Command or Action	Purpose
Step 2	**call-home** **Example:** `RP/0/RP0/CPU0:router (config) # call-home`	Enters Call Home configuration mode.
Step 3	**alert-group-configuration snapshot** **Example:** `RP/0/RP0/CPU0:router (config-call-home) # alert-group-configuration snapshot`	Enters snapshot configuration mode.
Step 4	**add-command** *"command string"* **Example:** `RP/0/RP0/CPU0:router (config-call-home-snapshot) # add-command "show ver"`	Adds the command to the snapshot alert group.

Configuring Anonymous Reporting

This task enables the user to configure an anonymous mode profile.

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **profile** *name*
4. **anonymous-reporting-only**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home** **Example:** `RP/0/RP0/CPU0:router (config) # call-home`	Enters Call Home configuration mode.
Step 3	**profile** *name* **Example:** `RP/0/RP0/CPU0:router (config-call-home) # profile ciscotac`	Enters the profile configuration mode.
Step 4	**anonymous-reporting-only** **Example:** `RP/0/RP0/CPU0:router (config-call-home-profile) # anonymous-reporting-only`	Enters anonymous mode. When **anonymous-reporting-only** is set, only inventory and test messages are sent.

What to do next

-

Configuring Call Home to use VRF

SUMMARY STEPS

1. **configure**
2. **call-home**
3. **vrf** *vrf-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home** **Example:** `RP/0/RP0/CPU0:router (config) # call-home`	Enters Call Home configuration mode.
Step 3	**vrf** *vrf-name* **Example:** `RP/0/RP0/CPU0:router (config) # vrf v1`	Configures call home for the specified VRF. VRF works only for the http transport method. It does not work for the email transport method.

Configuring Source Interface

This task enables the user to configure a source interface.

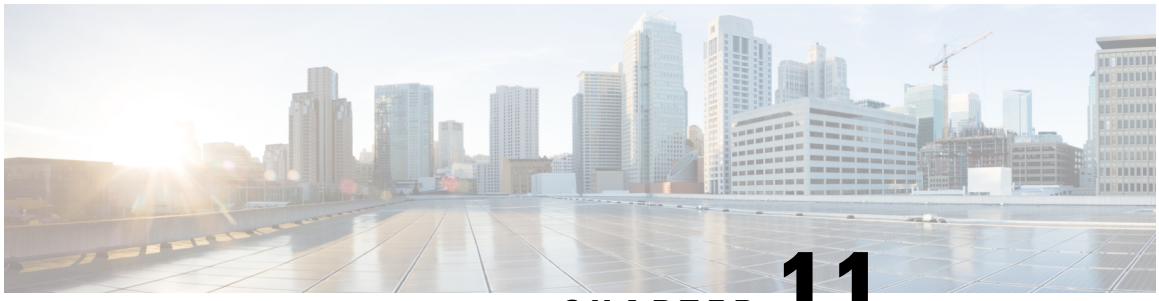
SUMMARY STEPS

1. **configure**
2. **call-home**
3. **source-interface** *type interface-path-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**call-home** **Example:** `RP/0/RP0/CPU0:router (config) # call-home`	Enters Call Home configuration mode.

	Command or Action	Purpose
Step 3	**source-interface** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router (config) # source-interface tengige 10.1.1.1`	Configures the source interface. **Note** Source-interface supports email and HTTP messages.



CHAPTER 11

Configuring Network Time Protocol

Network Time Protocol (NTP) is a protocol designed to time-synchronize devices within a network. Cisco IOS XR software implements NTPv4. NTPv4 retains backwards compatibility with the older versions of NTP, including NTPv3 and NTPv2 but excluding NTPv1, which has been discontinued due to security vulnerabilities.

The Cisco implementation of NTP supports both IPv4 and IPv6 addresses and supports VRF.

This module describes the tasks you need to implement NTP on the Cisco IOS XR software.

For more information about NTP on the Cisco IOS XR software and complete descriptions of the NTP commands listed in this module, see Related Documents, on page 274. To locate documentation for other commands that might appear in the course of running a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 29: Feature History for Implementing NTP on Cisco IOS XR Software

Release	Modification
Release 2.0	This feature was introduced.
Release 3.8.0	Support was added for IPv6 addresses, VRFs, multicast-based associations, and burst and iburst modes for poll-based associations.

This module contains the following topics:

Prerequisites for Implementing NTP on Cisco IOS XR Software

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Information About Implementing NTP

NTP synchronizes timekeeping among a set of distributed time servers and clients. This synchronization allows events to be correlated when system logs are created and other time-specific events occur.

NTP uses the User Datagram Protocol (UDP) as its transport protocol. All NTP communication uses Coordinated Universal Time (UTC). An NTP network usually receives its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of each other.

NTP uses the concept of a "stratum" to describe how many NTP "hops" away a machine is from an authoritative time source. A "stratum 1" time server typically has an authoritative time source (such as a radio or atomic clock, or a GPS time source) directly attached, a "stratum 2" time server receives its time via NTP from a "stratum 1" time server, and so on.

NTP avoids synchronizing to a machine whose time may not be accurate, in two ways. First, NTP never synchronizes to a machine that is not synchronized itself. Second, NTP compares the time reported by several machines and does not synchronize to a machine whose time is significantly different than the others, even if its stratum is lower. This strategy effectively builds a self-organizing tree of NTP servers.

The Cisco implementation of NTP does not support stratum 1 service; in other words, it is not possible to connect to a radio or atomic clock (for some specific platforms, however, you can connect a GPS time-source device). We recommend that time service for your network be derived from the public NTP servers available in the IP Internet.

If the network is isolated from the Internet, the Cisco implementation of NTP allows a machine to be configured so that it acts as though it is synchronized via NTP, when in fact it has determined the time using other means. Other machines can then synchronize to that machine via NTP.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software also allows UNIX-derivative servers to acquire the time directly from an atomic clock, which would subsequently propagate time information along to Cisco routers.

The communications between machines running NTP (known as *associations*) are usually statically configured; each machine is given the IP address of all machines with which it should form associations. Accurate timekeeping is made possible by exchanging NTP messages between each pair of machines with an association.

The Cisco implementation of NTP supports three ways that a networking device can obtain NTP time information on a network:

- By polling host servers
- By listening to NTP broadcasts
- By listening to NTP multicasts

In a LAN environment, NTP can be configured to use IP broadcast or multicast messages. As compared to polling, IP broadcast or multicast messages reduce configuration complexity, because each machine can simply be configured to send or receive broadcast or multicast messages. However, the accuracy of timekeeping is marginally reduced because the information flow is one-way only.

An NTP broadcast client listens for broadcast messages sent by an NTP broadcast server at a designated IPv4 address. The client synchronizes the local clock using the first received broadcast message.

An NTP multicast server periodically sends a message to a designated IPv4 or IPv6 local multicast group address. An NTP multicast client listens on this address for NTP messages.

The time kept on a machine is a critical resource, so we strongly recommend that you use the security features of NTP to avoid the accidental or malicious setting of incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

When multiple sources of time (VINES, hardware clock, manual configuration) are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Preventing Issues due to GPS Week Number Rollover (WNRO)

- If there are no GPS sources in the NTP source chain or server chain, there is no impact of GPS Week Number Rollover (WNRO).
- GPS WNRO affects only the system clock and not user traffic.
- Contact your GPS manufacturer to fix the GPS source for this condition.

To mitigate impact of GPS sources that are subject to GPS WNRO perform the following optional workarounds:

- If the GPS source has been identified to be a cause of potential disruption on April 6, 2019 (or after), configure ntp master in the Cisco that is device connected to this source, and its clock on the Stratum 1 device to preventively isolate it. This configuration enables the device to present its own clock for synchronization to downstream NTP clients.

Note The usage of ntp master command as mentioned above is only a workaround to this condition. Use this command until the GPS source-related conditions are resolved, and to prevent the distribution of incorrect clock values throughout the network.

- Configure multiple NTP servers (ideally 4, but more than 3) at Stratum 2 level of the network, to enable NTP clients at Stratum 2 level to get clock from more than one Stratum 1 server. This way, WNRO affected Stratum 1 servers are staged to be marked as ‘false ticker’ or ‘outlier’ clock sources as compared to other non-WNRO affected Stratum 1 servers.

How to Implement NTP

Configuring Poll-Based Associations

Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

You can configure the following types of poll-based associations between the router and other devices (which may also be routers):

- Client mode
- Symmetric active mode

The client and the symmetric active modes should be used when NTP is required to provide a high level of time accuracy and reliability.

When a networking device is operating in the client mode, it polls its assigned time serving hosts for the current time. The networking device then picks a host from all the polled time servers to synchronize with. Because the relationship that is established in this case is a client-host relationship, the host does not capture or use any time information sent by the local client device. This mode is most suited for file-server and workstation clients that are not required to provide any form of time synchronization to other local clients. Use the **server** command to individually specify the time-serving hosts that you want your networking device to consider synchronizing with and to set your networking device to operate in the client mode.

When a networking device is operating in the symmetric active mode, it polls its assigned time-serving hosts for the current time and it responds to polls by its hosts. Because this is a peer-to-peer relationship, the host also retains time-related information about the local networking device that it is communicating with. This mode should be used when there are several mutually redundant servers that are interconnected via diverse network paths. Most stratum 1 and stratum 2 servers on the Internet today adopt this form of network setup. Use the **peer** command to individually specify the time-serving hosts that you want your networking device to consider synchronizing with and to set your networking device to operate in the symmetric active mode.

When the router polls several other devices for the time, the router selects one device with which to synchronize.

Note To configure a peer-to-peer association between the router and another device, you must also configure the router as a peer on the other device.

You can configure multiple peers and servers, but you cannot configure a single IP address as both a peer and a server at the same time.

To change the configuration of a specific IP address from peer to server or from server to peer, use the **no** form of the **peer** or **server** command to remove the current configuration before you perform the new configuration. If you do not remove the old configuration before performing the new configuration, the new configuration does not overwrite the old configuration.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **server** *ip-address* [**version** *number*] [**key** *key-id*] [**minpoll** *interval*] [**maxpoll** *interval*] [**source** *type interface-path-id*] [**prefer**] [**burst**] [**iburst**]
4. **peer** *ip-address* [**version** *number*] [**key** *key-id*] [**minpoll** *interval*] [**maxpoll** *interval*] [**source** *type interface-path-id*] [**prefer**]
5. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

	Command or Action	Purpose
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.

Step 3	**server** *ip-address* [**version** *number*] [**key** *key-id*] [**minpoll** *interval*] [**maxpoll** *interval*] [**source** *type interface-path-id*] [**prefer**] [**burst**] [**iburst**]
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# server 172.16.22.44`
	`  minpoll 8 maxpoll 12`
		Forms a server association with another system. This step can be repeated as necessary to form associations with multiple devices.

Step 4	**peer** *ip-address* [**version** *number*] [**key** *key-id*] [**minpoll** *interval*] [**maxpoll** *interval*] [**source** *type interface-path-id*] [**prefer**]
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# peer 192.168.22.33`
	`  minpoll 8 maxpoll 12 source tengige 0/0/0/1`
		Forms a peer association with another system. This step can be repeated as necessary to form associations with multiple systems.
		Note To complete the configuration of a peer-to-peer association between the router and the remote device, the router must also be configured as a peer on the remote device.

Step 5	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# end`
	or
	`RP/0/RP0/CPU0:router(config-ntp)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		  • Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		  • Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		  • Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Broadcast-Based NTP Associates

In a broadcast-based NTP association, an NTP server propagates NTP broadcast packets throughout a network. Broadcast clients listen for the NTP broadcast packets propagated by the NTP server and do not engage in any polling.

Broadcast-based NTP associations should be used when time accuracy and reliability requirements are modest and if your network is localized and has a large number of clients (more than 20). Broadcast-based NTP associations also are recommended for use on networks that have limited bandwidth, system memory, or CPU resources. Time accuracy is marginally reduced in broadcast-based NTP associations because information flows only one way.

Use the **broadcast client** command to set your networking device to listen for NTP broadcast packets propagated through a network. For broadcast client mode to work, the broadcast server and its clients must be located on the same subnet. The time server that is transmitting NTP broadcast packets must be enabled on the interface of the given device using the **broadcast** command.

Use the **broadcast** command to set your networking device to send NTP broadcast packets.

Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. (Optional) **broadcastdelay** *microseconds*
4. **interface** *type interface-path-id*
5. **broadcast client**
6. **broadcast** [**destination** *ip-address*] [**key** *key-id*] [**version** *number*]
7. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.
Step 3	(Optional) **broadcastdelay** *microseconds* **Example:** `RP/0/RP0/CPU0:router(config-ntp)# broadcastdelay 5000`	Adjusts the estimated round-trip delay for NTP broadcasts.

	Command or Action	Purpose
Step 4	**interface** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router(config-ntp)# interface POS 0/1/0/0`	Enters NTP interface configuration mode.
Step 5	**broadcast client** **Example:** `RP/0/RP0/CPU0:router(config-ntp-int)# broadcast client`	Configures the specified interface to receive NTP broadcast packets. **Note** Go to next step to configure the interface to send NTP broadcast packets.
Step 6	**broadcast** [**destination** *ip-address*] [**key** *key-id*] [**version** *number*] **Example:** `RP/0/RP0/CPU0:router(config-ntp-int)# broadcast destination 10.50.32.149`	Configures the specified interface to send NTP broadcast packets. **Note** Go to previous step to configure the interface to receive NTP broadcast packets.

Step 7	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp-int)# end`
	or
	`RP/0/RP0/CPU0:router(config-ntp-int)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring Multicast-Based NTP Associations

Multicast-based NTP associations should be used when time accuracy and reliability requirements are modest and if your network is localized and has a large number of clients (more than 20). Multicast-based NTP associations also are recommended for use on networks that have limited bandwidth, system memory, or CPU resources.

When the router operates as an NTP multicast client, it listens for NTP multicast packets that are sent by an NTP multicast server to a designated IPv4 or IPv6 multicast group IP address.

When the router operates as an NTP multicast server, it sends NTP multicast messages to a designated IPv4 or IPv6 multicast group IP address.

Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **interface** *type interface-path-id* [**vrf** *vrf-name*]
4. **multicast client** [*ip-address*]
5. **multicast destination** *ip-address* [**key** *key-id*] [**version** *number*] [**ttl** *ttl*]
6. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.
Step 3	**interface** *type interface-path-id* [**vrf** *vrf-name*] **Example:** `RP/0/RP0/CPU0:router(config-ntp)# interface POS 0/1/0/0`	Enters NTP interface configuration mode.
Step 4	**multicast client** [*ip-address*] **Example:** `RP/0/RP0/CPU0:router(config-ntp-int)# multicast client`	Configures the specified interface to listen for NTP multicast packets on the specified IPv4 or IPv6 address. If no IP address is specified, the interface listens on the default IPv4 address 224.0.1.1. **Note** To configure the interface to send NTP multicast packets, go to Step 5, on page 260
Step 5	**multicast destination** *ip-address* [**key** *key-id*] [**version** *number*] [**ttl** *ttl*] **Example:**	Configures the specified interface to send NTP multicast packets to a specified IPv4 or IPv6 multicast group address. **Note** To configure the interface to listen for NTP multicast packets, go to Step 4, on page 260.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-ntp-int)# multicast destination 224.0.1.1`	

Step 6	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# end`
	or
	`RP/0/RP0/CPU0:router(config-ntp)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring NTP Access Groups

Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

The access list-based restriction scheme allows you to grant or deny certain access privileges to an entire network, a subnet within a network, or a host within a subnet. NTP communication consists of time requests and control queries. A *time request* is a request for time synchronization from an NTP server. A *control query* is a request for configuration information from an NTP server.

The access group options are scanned in the following order, from least restrictive to most restrictive:

1. **peer**—Allows time requests and NTP control queries and allows the system to synchronize itself to a system whose address passes the access list criteria.
2. **serve**—Allows time requests and NTP control queries, but does not allow the system to synchronize itself to a system whose address passes the access list criteria.
3. **serve-only**—Allows only time requests from a system whose address passes the access list criteria.
4. **query-only**—Allows only NTP control queries from a system whose address passes the access list criteria.

If the source IP address matches the access lists for more than one access type, the first type is granted. If no access groups are specified, all access types are granted to all systems. If any access groups are specified, only the specified access types are granted.

For details on NTP control queries, see RFC 1305 (NTP version 3).

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **access-group**{**peer** | **query-only** | **serve** | **serve-only**} *access-list-name*
4. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.
Step 3	**access-group**{**peer**	**query-only**	**serve**	**serve-only**} *access-list-name* **Example:** `RP/0/RP0/CPU0:router(config-ntp)# access-group peer access1`	Creates an access group and applies a basic IPv4 or IPv6 access list to it.

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# end`
	or
	`RP/0/RP0/CPU0:router(config-ntp)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

	Command or Action	Purpose
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring NTP Authentication

This task explains how to configure NTP authentication.

Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

The encrypted NTP authentication scheme should be used when a reliable form of access control is required. Unlike the access-list-based restriction scheme that is based on IP addresses, the encrypted authentication scheme uses authentication keys and an authentication process to determine if NTP synchronization packets sent by designated peers or servers on a local network are deemed as trusted, before the time information that it carries along is accepted.

The authentication process begins from the moment an NTP packet is created. A message authentication code (MAC) is computed using the MD5 Message Digest Algorithm and the MAC is embedded into an NTP synchronization packet. The NTP synchronization packet together with the embedded MAC and key number are transmitted to the receiving client. If authentication is enabled and the key is trusted, the receiving client computes the MAC in the same way. If the computed MAC matches the embedded MAC, the system is allowed to sync to the server that uses this key in its packets.

After NTP authentication is properly configured, your networking device only synchronizes with and provides synchronization to trusted time sources.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **authenticate**
4. **authentication-key** *key-number* **md5** [**clear** | **encrypted**] *key-name*
5. **trusted-key** *key-number*
6. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

	Command or Action	Purpose
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.
Step 3	**authenticate** **Example:** `RP/0/RP0/CPU0:router(config-ntp)# authenticate`	Enables the NTP authentication feature.
Step 4	**authentication-key** *key-number* **md5** [**clear** | **encrypted**] *key-name* **Example:** `RP/0/RP0/CPU0:router(config-ntp)# authentication-key 42 md5 clear key1`	Defines the authentication keys. • Each key has a key number, a type, a value, and, optionally, a name. Currently the only key type supported is **md5**.
Step 5	**trusted-key** *key-number* **Example:** `RP/0/RP0/CPU0:router(config-ntp)# trusted-key 42`	`Defines trusted authentication keys.` • If a key is trusted, this router only synchronizes to a system that uses this key in its NTP packets.

Step 6	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# end`
	or
	`RP/0/RP0/CPU0:router(config-ntp)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Disabling NTP Services on a Specific Interface

NTP services are disabled on all interfaces by default.

NTP is enabled globally when any NTP commands are entered. You can selectively prevent NTP packets from being received through a specific interface by turning off NTP on a given interface.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. Use one of the following commands:
   - **no interface** *type interface-path-id*
   - **interface** *type interface-path-id* **disable**
4. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.

Step 3	Use one of the following commands:
	• **no interface** *type interface-path-id*
	• **interface** *type interface-path-id* **disable**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# no interface pos 0/0/0/1`
	or
	`RP/0/RP0/CPU0:router(config-ntp)# interface POS 0/0/0/1 disable`
		Disables NTP services on the specified interface.

Step 4	Use one of the following commands: • **end** • **commit** **Example:** `RP/0/RP0/CPU0:router(config-ntp)# end`	Saves configuration changes. • When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`

	Command or Action	Purpose
	or `RP/0/RP0/CPU0:router(config-ntp)# commit`	• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the Source IP Address for NTP Packets

By default, the source IP address of an NTP packet sent by the router is the address of the interface through which the NTP packet is sent. Use this procedure to set a different source address.

Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **source** *type interface-path-id*
4. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.
Step 3	**source** *type interface-path-id* **Example:**	Configures an interface from which the IP source address is taken.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-ntp)# source POS 0/0/0/1`	**Note** This interface is used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the **source** keyword in the **peer** or **server** command shown in Configuring Poll-Based Associations, on page 255.

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# end`
	or
	`RP/0/RP0/CPU0:router(config-ntp)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Configuring the System as an Authoritative NTP Server

You can configure the router to act as an authoritative NTP server, even if the system is not synchronized to an outside time source.

Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **master** *stratum*
4. Use one of the following commands:
   • **end**

- **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.
Step 3	**master** *stratum* **Example:** `RP/0/RP0/CPU0:router(config-ntp)# master 9`	`Makes the router an authoritative NTP server.` **Note** Use the **master** command with caution. It is very easy to override valid time sources using this command, especially if a low stratum number is configured. Configuring multiple machines in the same network with the **master** command can cause instability in time keeping if the machines do not agree on the time.

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# end`
	or
	`RP/0/RP0/CPU0:router(config-ntp)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Updating the Hardware Clock

On devices that have hardware clocks (system calendars), you can configure the hardware clock to be periodically updated from the software clock. This is advisable for devices using NTP, because the time and date on the software clock (set using NTP) is more accurate than the hardware clock. The time setting on the hardware clock has the potential to drift slightly over time.

Note No specific command enables NTP; the first NTP configuration command that you issue enables NTP.

SUMMARY STEPS

1. **configure**
2. **ntp**
3. **update-calendar**
4. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ntp** **Example:** `RP/0/RP0/CPU0:router(config)# ntp`	Enters NTP configuration mode.
Step 3	**update-calendar** **Example:** `RP/0/RP0/CPU0:router(config-ntp)# update-calendar`	Configures the router t o update its system calendar from the software clock at periodic intervals.

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-ntp)# end`
	or
	`RP/0/RP0/CPU0:router(config-ntp)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.

	Command or Action	Purpose
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Verifying the Status of the External Reference Clock

This task explains how to verify the status of NTP components.

Note The commands can be entered in any order.

SUMMARY STEPS

1. **show ntp associations** [**detail**] [**location** *node-id*]
2. **show ntp status** [**location** *node-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ntp associations** [**detail**] [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show ntp associations`	Displays the status of NTP associations.
Step 2	**show ntp status** [**location** *node-id*] **Example:** `RP/0/RP0/CPU0:router# show ntp status`	Displays the status of NTP.

Examples

The following is sample output from the **show ntp associations** command:

```
RP/0/RP0/CPU0:router# show ntp associations

Tue Oct  7 11:22:46.839 JST

      address         ref clock     st  when  poll reach  delay  offset    disp
```

```
*~192.168.128.5    10.81.254.131     2     1    64  377    7.98  -0.560   0.108
+~dead:beef::2 vrf testAA
                   171.68.10.80      3    20    64  377    6.00  -2.832   0.046
* sys_peer, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

The following is sample output from the **show ntp status** command:

```
RP/0/RP0/CPU0:router# show ntp status

Tue Oct  7 11:22:54.023 JST

Clock is synchronized, stratum 3, reference is 192.168.128.5
nominal freq is 1000.0000 Hz, actual freq is 1000.2725 Hz, precision is 2**24
reference time is CC95463C.9B964367 (11:21:48.607 JST Tue Oct  7 2008)
clock offset is -1.738 msec, root delay is 186.050 msec
root dispersion is 53.86 msec, peer dispersion is 0.09 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is -0.0002724105 s/s
system poll interval is 64, last update was 66 sec ago
```

Configuration Examples for Implementing NTP

Configuring Poll-Based Associations: Example

The following example shows an NTP configuration in which the router's system clock is configured to form a peer association with the time server host at IP address 192.168.22.33, and to allow the system clock to be synchronized by time server hosts at IP address 10.0.2.1 and 172.19.69.1:

```
ntp
  server 10.0.2.1 minpoll 5 maxpoll 7
  peer 192.168.22.33

  server 172.19.69.1
```

Configuring Broadcast-Based Associations: Example

The following example shows an NTP client configuration in which interface 0/2/0/0 is configured to receive NTP broadcast packets, and the estimated round-trip delay between an NTP client and an NTP broadcast server is set to 2 microseconds:

```
ntp
  interface tengige 0/2/0/0
    broadcast client
    exit
  broadcastdelay 2
```

The following example shows an NTP server configuration where interface 0/2/0/2 is configured to be a broadcast server:

```
ntp
  interface tengige 0/2/0/2
```

```
    broadcast
```

Configuring Multicast-Based Associations: Example

The following example shows an NTP multicast client configuration where 10-Gigabit Ethernet interface 0/1/1/0 is configured to be a multicast client and to join the default multicast group (IPv4 address 224.0.1.1):

```
ntp interface TenGigE 0/1/1/0
  multicast client
```

The following example shows an NTP multicast server configuration where 10-Gigabit Ethernet interface 0/1/1/0 is configured to be a multicast server:

```
ntp interface TenGigE 0/1/1/0
  multicast destination 224.0.1.1
```

Configuring NTP Access Groups: Example

The following example shows a NTP access group configuration where the following access group restrictions are applied:

- Peer restrictions are applied to IP addresses that pass the criteria of the access list named peer-acl.
- Serve restrictions are applied to IP addresses that pass the criteria of access list named serve-acl.
- Serve-only restrictions are applied to IP addresses that pass the criteria of the access list named serve-only-acl.
- Query-only restrictions are applied to IP addresses that pass the criteria of the access list named query-only-acl.

```
ntp
  peer 10.1.1.1
  peer 10.1.1.1
  peer 10.2.2.2
  peer 10.3.3.3
  peer 10.4.4.4
  peer 10.5.5.5
  peer 10.6.6.6
  peer 10.7.7.7
  peer 10.8.8.8
  access-group peer peer-acl
  access-group serve serve-acl
  access-group serve-only serve-only-acl
  access-group query-only query-only-acl
  exit
ipv4 access-list peer-acl
  10 permit ip host 10.1.1.1 any
  20 permit ip host 10.8.8.8 any
  exit
ipv4 access-list serve-acl
  10 permit ip host 10.4.4.4 any
  20 permit ip host 10.5.5.5 any
  exit
ipv4 access-list query-only-acl
```

```
  10 permit ip host 10.2.2.2 any
  20 permit ip host 10.3.3.3 any
  exit
ipv4 access-list serve-only-acl
  10 permit ip host 10.6.6.6 any
  20 permit ip host 10.7.7.7 any
  exit
```

Configuring NTP Authentication: Example

The following example shows an NTP authentication configuration. In this example, the following is configured:

- NTP authentication is enabled.
- Two authentication keys are configured (key 2 and key 3).
- The router is configured to allow its software clock to be synchronized with the clock of the peer (or vice versa) at IP address 10.3.32.154 using authentication key 2.
- The router is configured to allow its software clock to be synchronized with the clock by the device at IP address 10.32.154.145 using authentication key 3.
- The router is configured to synchronize only to systems providing authentication key 3 in their NTP packets.

```
ntp
  authenticate
  authentication-key 2 md5 encrypted 06120A2D40031D1008124
  authentication-key 3 md5 encrypted 1311121E074110232621
  trusted-key 3
  server 10.3.32.154 key 3
  peer 10.32.154.145 key 2
```

Disabling NTP on an Interface: Example

The following example shows an NTP configuration in which 0/2/0/0 interface is disabled:

```
ntp
  interface tengige 0/2/0/0
    disable
    exit
  authentication-key 2 md5 encrypted 06120A2D40031D1008124
  authentication-key 3 md5 encrypted 1311121E074110232621
  authenticate
  trusted-key 3
  server 10.3.32.154 key 3
  peer 10.32.154.145 key 2
```

Configuring the Source IP Address for NTP Packets: Example

The following example shows an NTP configuration in which Ethernet management interface 0/0/CPU0/0 is configured as the source address for NTP packets:

```
ntp
  authentication-key 2 md5 encrypted 06120A2D40031D1008124
  authentication-key 3 md5 encrypted 1311121E074110232621
  authenticate
  trusted-key 3
  server 10.3.32.154 key 3
  peer 10.32.154.145 key 2
  source MgmtEth0/0/CPU0/0
```

Configuring the System as an Authoritative NTP Server: Example

The following example shows a NTP configuration `in which the router` is configured to use its own NTP master clock to synchronize with peers when an external NTP source becomes unavailable:

```
ntp
  master 6
```

Updating the Hardware Clock: Example

The following example shows an NTP configuration in which `the router is configured to update its hardware clock from the software clock at periodic intervals:`

```
ntp
  server 10.3.32.154
  update-calendar
```

Additional References

The following sections provide references related to implementing NTP on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Cisco IOS XR clock commands	*Clock Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*
Cisco IOS XR NTP commands	*NTP Commands on* module of *System Management Command Reference for isco CRS Routers*
Information about getting started with Cisco IOS XR Software	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*
Cisco IOS XR master command index	*Cisco IOS XR Commands Master List for the Cisco CRS Router*

Related Topic	Document Title
Information about user groups and task IDs	*Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

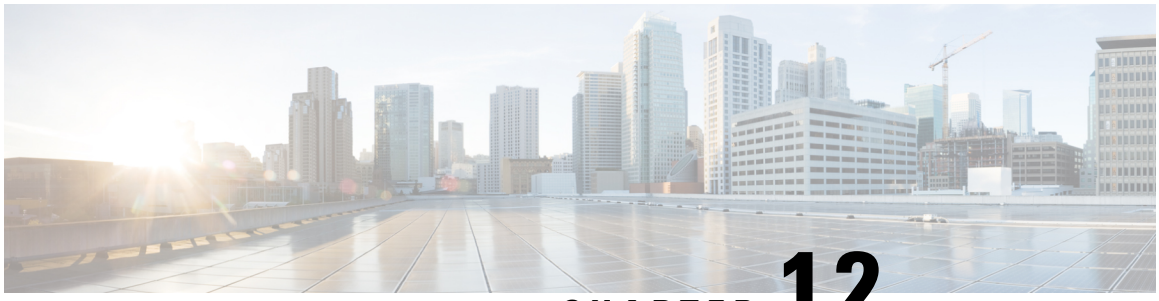
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
RFC 1059	*Network Time Protocol, Version 1: Specification and Implementation*
RFC 1119	*Network Time Protocol, Version 2: Specification and Implementation*
RFC 1305	*Network Time Protocol, Version 3: Specification, Implementation, and Analysis*

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 12

Configuring Network Configuration Protocol

This module provides details of the Network Configuration Protocol. For relevant commands, see *System Security Command Reference for Cisco CRS Routers*.

Release	Modification
Release 5.3.0	This feature was introduced.
Release 5.3.1	Support extended for more Yang models.
Release 6.0	Support extended for the Netconf subsystem configuration to be vrf aware. The configuration of the netconf port is no longer sufficient to start the Netconf subsystem support. At least one vrf needs to be configured. The configuration of the port is now optional.

The Network Configuration Protocol

The Network Configuration Protocol (Netconf) provides mechanisms to install, manipulate, and delete the configuration of network devices. It uses an Extensible Markup Language (XML)-based data encoding for the configuration data as well as the protocol messages. Yang is a data modeling language used with Netconf.

Netconf uses a simple RPC-based (Remote Procedure Call) mechanism to facilitate communication between a client and a server. The client can be a script or application typically running as part of a network manager. The server is typically a network device.

The configuration of features need not be done the traditional way (using CLIs), the client application (controller) reads the Yang model and communicates with the Netconf server (IOS XR) accordingly.

Netconf Sessions and Operations

A Netconf session is the logical connection between a network configuration application and a network device. A device should be capable of supporting multiple sessions and atleast one Netconf session.

Characteristics of a netconf session:

- Netconf is connection-oriented - SSH is the underlying transport.
- The netconf client establishes session with the server.
- Netconf sessions are established with the *hello* message. Features and capabilities are announced.
- Sessions can be terminated using the *close* or *kill* messages.

Basic Netconf operations:

- Get configuration <get-config>
- Get all information <get>
- Edit configuration <edit-config>
- Copy configuration <copy-config>

Note <copy-config> does not support source attribute with “data store” at present.

- <lock>, <unlock>
- <kill-session>
- <close-session>
- Commit configuration <commit>

The Yang data model

Each feature has a defined Yang Model which is synthesized from the schemas. A model is published in a tree format and includes:

- Top level nodes and their subtrees
- Subtrees that augment nodes in other yang models

```
Example: The aaa Yang model
module: Cisco-IOS-XR-aaa-lib-cfg
   +--rw aaa
      +--rw accountings
      |  +--rw accounting* [type listname]
      |     +--rw type                xr:Cisco-ios-xr-string
      |     +--rw listname            xr:Cisco-ios-xr-string
      |     +--rw rp-failover?        Aaa-accounting-rp-failover
      |     +--rw broadcast?          Aaa-accounting-broadcast
      |     +--rw type-xr?            Aaa-accounting
      |     +--rw method*             Aaa-method
      |     +--rw server-group-name*  string
```

```
+--rw authorizations
|  +--rw authorization* [type listname]
|     +--rw type                 xr:Cisco-ios-xr-string
|     +--rw listname             xr:Cisco-ios-xr-string
|     +--rw method*              Aaa-method
|     +--rw server-group-name*   string
+--rw accounting-update!
|  +--rw type                 Aaa-accounting-update
|  +--rw periodic-interval?   uint32
+--rw authentications
   +--rw authentication* [type listname]
      +--rw type                 xr:Cisco-ios-xr-string
      +--rw listname             xr:Cisco-ios-xr-string
      +--rw method*              Aaa-method
      +--rw server-group-name*   string
```

Advantages of using the Yang model are:

- Yang supports programmatic interfaces.
- Yang supports simplified network management applications.
- Yang supports interoperability that provides a standard way to model management data.

Netconf and Yang

The workflow displayed here, will help the user to understand how Netconf-Yang can configure and control the network with minimal user intervention. The required components:

- Cisco Router (ASR9000 series or CRS) with Netconf capability
- Netconf Client Application with connection to the router

S. No.	Device / component	Action
1	Cisco router (ASR 9000 or CRS router)	Login/ access the router.
2	Cisco router	Prerequisites for enabling Netconf. • k9sec pie must be installed. • Crypto keys must be generated.
3	Cisco router	Enable Netconf agent. Use the **netconf-yang agent ssh** and **ssh server netconf** command. The port can be selected. By default, it is set as 830.
4	Cisco router	Yang models are a part of the software image. The models can be retrieved from the router , using the <get-schema> operation.

S. No.	Device / component	Action
5	Netconf client (application) The application can be on any standalone application or a SDN controller supporting Netconf	Installs and processes the Yang models. The client can offer a list of supported yang models; else the user will have to browse and locate the required yang file. There is a yang model file for each configuration module; for instance if the user wants to configure CDP , the relevant yang model is Cisco-IOS-XR-cdp-cfg **Note** Refer the table which lists all the supported yang models. Supported Yang Models , on page 280
5	Netconf client	Sends Netconf operation request over SSH to the router. A configuration request could include Yang-based XML data to the router. Currently, SSH is the only supported transport method.
6	Cisco router	Understands the Yang-based XML data and the network is configured accordingly (in case of configuration request from the client).
		The interactions between the client and the router happens until the network is configured as desired.

Supported Yang Models

The Yang models can be downloaded from a prescribed location (ftp server) or can also be retrieved directly from the router using the get-schema operation.

For a feature, separate Yang models are available for configuring the feature and to get operational statistics (show commands). The **-cfg.yang** suffix denotes configuration and **-oper*.yang** is for operational data statistics. In some cases, **-oper** is followed by **-sub**, indicating that a submodule(s) is available.

For a list of supported Yang models, see https://github.com/YangModels/yang/tree/master/vendor/cisco/xr

Denial of Services Defence for Netconf-Yang

In case of a DoS (Denial of Service) attack on Netconf, wherein, Netconf receives numerous requests in a short span of time, the router may become irresponsive if Netconf consumes most of the bandwidth or CPU processing time. This can be prevented, by limiting the traffic directed at the Netconf agent. This is achieved using the **netconf-yang agent rate-limit** and **netconf-yang agent session** commands.

If rate-limit is set, the Netconf processor measures the incoming traffic from the SSH server. If the incoming traffic exceeds the set rate-limit, the packets are dropped.

If session-limit is set, the Netconf processor checks for the number of open sessions. If the number of current sessions is greater than or equal to, the set limit, no new sessions are opened.

Session idle- timeout and absolute-timeout also prevent DoS attacks. The Netconf processor closes the sessions, even without user input or intervention, as soon at the time out session is greater than or equal to the set time limit.

The relevant commands are discussed in detail, in the *System Security Command Reference for Cisco CRS Routers*

Enabling NETCONF over SSH

This task enables NETCONF over SSH. SSH is currently the only supported transport method .

If the client supports, Netconf over ssh can utilize the multi-channeling capabilities of IOS XR ssh server. For additional details about Multi-channeling in SSH, see *Implementing Secure Shell* in *System Security Configuration Guide*.

Prerequisites:

- k9sec pie must be installed, otherwise the port configuration for the netconf ssh server cannot be completed. (The Netconf subsystem for SSH, as well as, SSH cannot be configured without the k9sec pie.)
- Crypto keys must be generated prior to this configuration.
- The Netconf-YANG feature is packaged in the mgbl pie, which must be installed before enabling the Netconf-YANG agent.

SUMMARY STEPS

1. **configure**
2. **netconf-yang agent ssh**
3. **ssh server netconf** [**vrf** *vrf-name* [**ipv4 access-list***ipv4 access list name*] [**ipv6 access-list** *ipv6 access list name*]]
4. **ssh server netconf port** *port-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**netconf-yang agent ssh** **Example:** `RP/0/RP0/CPU0:router (config) # netconf agent ssh`	Enables NETCONF agent over SSH connection. After NETCONF is enabled, the Yang model in the controllcker, can configure the relevant models. **Note** The Yang models can be retrieved from the router via NETCONF <get-schema> operation.
Step 3	**ssh server netconf** [**vrf** *vrf-name* [**ipv4 access-list***ipv4 access list name*] [**ipv6 access-list** *ipv6 access list name*]] **Example:** `RP/0/RP0/CPU0:router (config) # ssh server netconf vrf netconfvrf ipv4 access-list InternetFilter`	Brings up the netconf subsytem support with SSH server using a specified VRF of up to 32 characters. If no VRF is specified, the default VRF is used. To stop the SSH server from receiving any further connections for the specified VRF, use the **no** form of this command.

	Command or Action	Purpose
		Optionally ACLs for IPv4 and IPv6 can be used to restrict access to the netconf subsystem of the ssh server before the port is opened. **Note** The netconf subsystem support with SSH server can be configured for use with multiple VRFs .
Step 4	**ssh server netconf port** *port-number* **Example:** `RP/0/RP0/CPU0:router (config) # ssh server netconf port 830`	Configures a port for the netconf ssh server. This command is optional. If no port is specified, port 830 is uses by default. **Note** 830 is the IANA-assigned TCP port for NETCONF over SSH, but it can be changed using this command.

What to do next

The **show netconf-yang statistics** command and **show netconf-yang clients** command can be used to verify the configuration details of the netconf agent.

The **clear netconf-yang agent session** command clears the specified Netconf session (on the Netconf server side).

Examples: Netconf over SSH

This section illustrates some examples relevant to Netconf:

Enabling netconf-yang for ssh transport and netconf subsystem for default vrf with default port (830)

```
config
netconf-yang agent ssh
ssh server netconf vrf default
  !
!
```

Enabling netconf-yang for ssh transport and netconf subsystem for vrf *green* and vrf *red* with netconf port (831)

```
config
netconf-yang agent ssh
!
ssh server netconf vrf green
ssh server netconf vrf red
ssh server netconf port 831
  !
!
```

Show command outputs

```
show netconf-yang statistics
Summary statistics        requests|             total time|   min time per request|   max
time per request|   avg time per request|
other                              0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
close-session                      4|       0h  0m  0s   3ms|       0h  0m  0s   0ms|
```

```
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
kill-session                        0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get-schema                          0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get                                 0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s
get-config                          1|       0h  0m  0s   1ms|       0h  0m  0s   1ms|
 0h  0m  0s   1ms|       0h  0m  0s   1ms|
edit-config                         3|       0h  0m  0s   2ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
commit                              0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
cancel-commit                       0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
lock                                0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
unlock                              0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
discard-changes                     0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
validate                            0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|

show netconf-yang clients
client session ID|   NC version|    client connect time|        last OP time|        last
OP type|    <lock>|
22969|                 1.1|         0d  0h  0m  2s|             11:11:24|
close-session|        No|
15389|                1.1|        0d  0h  0m  1s|            11:11:25|         get-config|
          No|
```

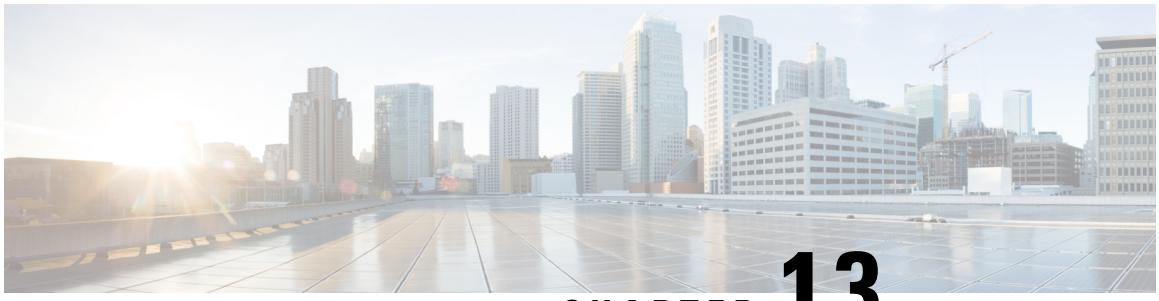
Additional Reference

Table 30: Related Documents

Related Topic	Document Title
Netconf-Yang	For related commands, see *System Security Command Reference for Cisco CRS Routers*

Table 31: Standards

Component	RFCs
YANG	6020
NETCONF	6241
NETCONF over SSH	6242



CHAPTER 13

Configuring Object Tracking

This module describes the configuration of object tracking on your Cisco IOS XR network. For complete descriptions of the commands listed in this module, see Related Documents, on page 296. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 32: Feature History for Implementing Object Tracking

Release	Modification
Release 4.2.1	This feature was introduced.

This module contains the following topics:

Prerequisites for Implementing Object Tracking

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Information About Object Tracking

Object tracking is a mechanism for tracking an object to take any client action on another object as configured by the client. The object on which the client action is performed may not have any relationship to the tracked objects. The client actions are performed based on changes to the properties of the object being tracked.

You can identify each tracked object by a unique name that is specified by the track command in the configuration mode.

The tracking process periodically polls the tracked object and reports any changes to its state. The state of the tracked objects can be up or down. The polling occurs either immediately or after a delay of a configured period.

You can also track multiple objects by a list. You can use a flexible method for combining objects with Boolean logic. This functionality includes:

- **Boolean AND function—**When a tracked list has been assigned a Boolean AND function, each object that is defined within a subset must be in an "up" state. This condition enables the tracked object to be in the "up" state.
- **Boolean OR function—**When the tracked list has been assigned a Boolean OR function, at least one object that is defined within a subset must also be in an "up" state. This condition enables the tracked object to be in the "up" state.

How to Implement Object Tracking

This section describes the various object tracking procedures.

Tracking the Line Protocol State of an Interface

Perform this task in global configuration mode to track the line protocol state of an interface.

A tracked object is considered up when a line protocol of the interface is up.

After configuring the tracked object, you may associate the interface whose state should be tracked and specify the number of seconds to wait before the tracking object polls the interface for its state.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type line-protocol state**
4. **interface** *type interface-path-id*
5. **exit**
6. (Optional) **delay** {**up** *seconds* | **down** *seconds*}
7. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**track** *track-name* **Example:** `RP/0/RP0/CPU0:router(config)# track track1`	Enters track configuration mode. • *track-name*—Specifies a name for the object to be tracked.

	Command or Action	Purpose
Step 3	**type line-protocol state** **Example:** `RP/0/RP0/CPU0:router(config-track)# type line-protocol state`	Creates a track based on the line protocol of an interface.
Step 4	**interface** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router(config-track-line-prot)# interface atm 0/2/0/0.1`	Specifies the interface to track the protocol state. • *type*—Specifies the interface type. For more information, use the question mark (?) online help function. • *interface-path-id*—Identifies a physical interface or a virtual interface. **Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router. **Note** The loopback and null interfaces are always in the up state and, therefore, cannot be tracked.
Step 5	**exit** **Example:** `RP/0/RP0/CPU0:router(config-track-line-prot)# exit`	Exits the track line protocol configuration mode.
Step 6	(Optional) **delay** {**up** *seconds* | **down** *seconds*} **Example:** `RP/0/RP0/CPU0:router(config-track)# delay up 10`	Schedules the delay that can occur between tracking whether the object is up or down.

Step 7	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-track)# end`
	or
	`RP/0/RP0/CPU0:router(config-track)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.

	Command or Action	Purpose
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Tracking IP Route Reachability

When a host or a network goes down on a remote site, routing protocols notify the router and the routing table is updated accordingly. The routing process is configured to notify the tracking process when the route state changes due to a routing update.

A tracked object is considered up when a routing table entry exists for the route and the route is accessible.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type route reachability**
4. Use one of the following commands:
   - **vrf** *vrf-table-name*
   - **route ipv4** *IP-prefix/mask*
5. **exit**
6. (Optional) **delay** {**up** *seconds* | **down** *seconds*}
7. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**track** *track-name* **Example:** `RP/0/RP0/CPU0:router(config)# track track1`	Enters track configuration mode. • *track-name*—Specifies a name for the object to be tracked.
Step 3	**type route reachability** **Example:** `RP/0/RP0/CPU0:router(config-track)# type route reachability vrf internet`	Configures the routing process to notify the tracking process when the state of the route changes due to a routing update.
Step 4	Use one of the following commands: • **vrf** *vrf-table-name* • **route ipv4** *IP-prefix/mask* **Example:**	Configures the type of IP route to be tracked, which can consist of either of the following, depending on your router type: • *vrf-table-name*—A VRF table name. • *IP-prefix/mask*—An IP prefix consisting of the network and subnet mask (for example, 10.56.8.10/16).

	Command or Action	Purpose

	`RP/0/RP0/CPU0:router(config-track-route)# vrf vrf-table-4`
	or
	`RP/0/RP0/CPU0:router(config-track-route)# route ipv4 10.56.8.10/16`	

Step 5	**exit** **Example:** `RP/0/RP0/CPU0:router(config-track-line-prot)# exit`	Exits the track line protocol configuration mode.
Step 6	(Optional) **delay** {**up** *seconds* | **down** *seconds*} **Example:** `RP/0/RP0/CPU0:router(config-track)# delay up 10`	Schedules the delay that can occur between tracking whether the object is up or down.
Step 7	**commit**	

Building a Track Based on a List of Objects

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a Boolean expression to determine the state of the list.

A tracked list contains one or more objects. The Boolean expression enables two types of calculations by using either AND or OR operators. For example, when tracking two interfaces, using the AND operator, up means that *both* interfaces are up, and down means that *either* interface is down.

Note An object must exist before it can be added to a tracked list.

The NOT operator is specified for one or more objects and negates the state of the object.

After configuring the tracked object, you must associate the interface whose state should be tracked and you may optionally specify the number of seconds to wait before the tracking object polls the interface for its state.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list boolean** { **and** | **or** }
4. **object** *object-name* [**not**]
5. **exit**
6. (Optional) **delay** {**up** *seconds* | **down** *seconds*}
7. Use one of the following commands:
   - **end**

- **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**track** *track-name* **Example:** `RP/0/RP0/CPU0:router(config)# track track1`	Enters track configuration mode. • *track-name*—Specifies a name for the object to be tracked.
Step 3	**type list boolean** { **and** | **or** } **Example:** `RP/0/RP0/CPU0:router(config-track-list)# type list boolean and`	Configures a Boolean list object and enters track list configuration mode. • **boolean**—Specifies that the state of the tracked list is based on a Boolean calculation. • **and**—Specifies that the list is up if all objects are up, or down if one or more objects are down. For example when tracking two interfaces, up means that both interfaces are up, and down means that either interface is down. • **or**—Specifies that the list is up if at least one object is up. For example, when tracking two interfaces, up means that either interface is up, and down means that both interfaces are down.
Step 4	**object** *object-name* [**not**] **Example:** `RP/0/RP0/CPU0:router(config-track-list)# object 3 not`	Specifies the object to be tracked by the list • *obect-name*—Name of the object to track. • **not**—Negates the state of the object.
Step 5	**exit** **Example:** `RP/0/RP0/CPU0:router(config-track-line-prot)# exit`	Exits the track line protocol configuration mode.
Step 6	(Optional) **delay** {**up** *seconds* | **down** *seconds*} **Example:** `RP/0/RP0/CPU0:router(config-track)# delay up 10`	Schedules the delay that can occur between tracking whether the object is up or down.
Step 7	Use one of the following commands: • **end** • **commit** **Example:** `RP/0/RP0/CPU0:router(config-track)# end` or	Saves configuration changes. • When you issue the **end** command, the system prompts you to commit changes: `Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-track)# commit`	• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode. • Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes. • Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Building a Track Based on a List of Objects - Threshold Percentage

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a threshold percentage to determine the state of the list.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list threshold percentage**
4. **object** *object-name*
5. **threshold percentage up** *percentage* **down** *percentage*
6. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**track** *track-name* **Example:** `RP/0/RP0/CPU0:router(config)# track track1`	Enters track configuration mode. • *track-name*—Specifies a name for the object to be tracked.
Step 3	**type list threshold percentage** **Example:**	Configures a track of type threshold percentage list.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-track-list)# type list threshold percentage`	

Step 4	**object** *object-name*
	Example:
	`RP/0/RP0/CPU0:router(config-track-list-threshold)# object 1`
	`RP/0/RP0/CPU0:router(config-track-list-threshold)# object 2`
	`RP/0/RP0/CPU0:router(config-track-list-threshold)# object 3`
	`RP/0/RP0/CPU0:router(config-track-list-threshold)# object 4`
		Configures object 1, object 2, object 3 and object 4 as members of track type track1.

Step 5	**threshold percentage up** *percentage* **down** *percentage* **Example:** `RP/0/RP0/CPU0:router(config-track-list-threshold)# threshold percentage up 50 down 33`	Configures the percentage of objects that need to be UP or DOWN for the list to be considered UP or Down respectively. For example, if object 1, object 2, and object 3 are in the UP state and object 4 is in the DOWN state, the list is considered to be in the UP state.

Step 6	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-track)# end`
	or
	`RP/0/RP0/CPU0:router(config-track)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Building a Track Based on a List of Objects - Threshold Weight

Perform this task in the global configuration mode to create a tracked list of objects (which, in this case, are lists of interfaces or prefixes) using a threshold weight to determine the state of the list.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type list threshold weight**
4. **object** *object-name* **weight** *weight*
5. **threshold weight up** *weight* **down** *weight*
6. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**track** *track-name* **Example:** `RP/0/RP0/CPU0:router(config)# track track1`	Enters track configuration mode. • *track-name*—Specifies a name for the object to be tracked.
Step 3	**type list threshold weight** **Example:** `RP/0/RP0/CPU0:router(config-track-list)# type list threshold weight`	Configures a a track of type, threshold weighted list.

Step 4	**object** *object-name* **weight** *weight*
	Example:
	`RP/0/RP0/CPU0:router(config-track-list-threshold)# object 1 weight 10`
	`RP/0/RP0/CPU0:router(config-track-list-threshold)# object 2 weight 5`
	`RP/0/RP0/CPU0:router(config-track-list-threshold)# object 3 weight 3`
		Configures object 1, object 2 and object 3 as members of track t1 and with weights 10, 5 and 3 respectively.

Step 5	**threshold weight up** *weight* **down** *weight* **Example:** `RP/0/RP0/CPU0:router(config-track-list-threshold)# threshold weight up 10 down 5`	Configures the range of weights for the objects that need to be UP or DOWN for the list to be considered UP or DOWN respectively. In this example, the list is considered to be in the DOWN state because objects 1 and 2 are in the UP state and the cumulative weight is 15 (not in the 10-5 range).
Step 6	Use one of the following commands:	Saves configuration changes.

	Command or Action	Purpose

	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-track)# end`
	or
	`RP/0/RP0/CPU0:router(config-track)# commit`
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Tracking IPSLA Reachability

Use this task to enable the tracking of the return code of IP service level agreement (SLA) operations.

SUMMARY STEPS

1. **configure**
2. **track** *track-name*
3. **type rtr** *ipsla-no* **reachability**
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**track** *track-name* **Example:** `RP/0/RP0/CPU0:router(config)# track t1`	Enters track configuration mode.

	Command or Action	Purpose
Step 3	**type rtr** *ipsla-no* **reachability** **Example:** `RP/0/RP0/CPU0:router(config-track)# type rtr 100 reachability`	Specifies the IP SLA operation ID to be tracked for reachability. Values for the *ipsla-no* can range from1 to 2048.
Step 4	**commit**	

Configuring IPSLA Tracking: Example

This example shows the configuration of IPSLA tracking:

```
RP/0/RP0/CPU0:router(config)# track track1
RP/0/RP0/CPU0:router(config-track)# type rtr 1 reachability
RP/0/RP0/CPU0:router(config-track)# delay up 5
RP/0/RP0/CPU0:router(config-track)# delay down 10
```

Configuration Examples for Configuring Object Tracking

Configuring IPSLA Tracking: Example

This example shows the configuration of IPSLA tracking, including the ACL and IPSLA configuration:

ACL configuration:

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list abf-track
RP/0/RP0/CPU0:router(config-ipv4-acl)# 10 permit any any nexthop track track1 1.2.3.4
```

Object tracking configuration:

```
RP/0/RP0/CPU0:router(config)# track track1
RP/0/RP0/CPU0:router(config-track)# type rtr 1 reachability
RP/0/RP0/CPU0:router(config-track)# delay up 5
RP/0/RP0/CPU0:router(config-track)# delay down 10
```

IPSLA configuration:

```
RP/0/RP0/CPU0:router(config)# ipsla
RP/0/RP0/CPU0:router(config-ipsla)# operation 1
RP/0/RP0/CPU0:router(config-ipsla-op)# type icmp echo
RP/0/RP0/CPU0:router(config-ipsla-icmp-echo)# source address 2.3.4.5
RP/0/RP0/CPU0:router(config-ipsla-icmp-echo)# destination address 1.2.3.4
RP/0/RP0/CPU0:router(config-ipsla-icmp-echo)# frequency 60
RP/0/RP0/CPU0:router(config-ipsla-icmp-echo)# exit
RP/0/RP0/CPU0:router(config-ipsla-op)# exit
RP/0/RP0/CPU0:router(config-ipsla)# schedule operation 1
RP/0/RP0/CPU0:router(config-ipsla-sched)# start-time now
```

```
RP/0/RP0/CPU0:router(config-ipsla-sched)# life forever
```

Additional References

The following sections provide references related to implementing object tracking for IPSec network security.

Related Documents

Related Topic	Document Title
IP SLA configuration information	*Implementing IP Service Level Agreements on the Cisco IOS XR Software* module in *System Monitoring Configuration Guide for Cisco CRS Routers*
IP SLA commands	*IP Service Level Agreement Commands on the Cisco IOS XR Software* module in *System Monitoring Command Reference for Cisco CRS Routers*
Object tracking commands	*Object Tracking Commands on the Cisco IOS XR Software* module in *System Management Command Reference for isco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

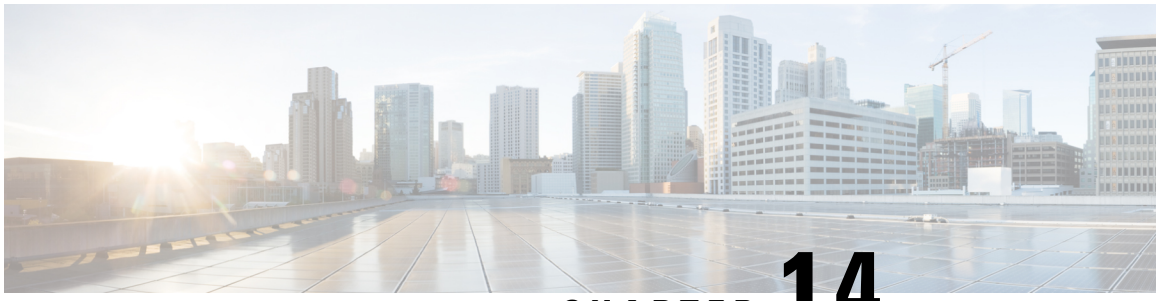
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
RFC 2401	*Security Architecture for the Internet Protocol*

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 14

Process Placement

This module describes conceptual information and configuration tasks for process placement on your router.

Process Placement on Cisco IOS XR software balances application processes between the available route processors (RPs) and distributed route processors (DRPs) based on memory usage and other criteria. Use the procedures described in this document to reoptimize the placement of processes, or override the default placement policies.

For complete descriptions of the process placement commands listed in this module, see Related Documents, on page 312. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 33: Feature History for Configuring Cisco IOS XR Process Placement

Release	Modification
Release 3.3.0	This feature was introduced.

This module contains the following topics:

Prerequisites for Configuring Cisco IOS XR Process Placement

Note Only processes that are identified in Cisco IOS XR software as placeable can be controlled through process placement configuration. Nonplaceable processes are not affected by placement policy. To learn the processes that are placeable, issue the **show placement program all** command.

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Information About Cisco IOS XR Process Placement

What Is a Process?

To achieve high availability and performance, the Cisco IOS XR software is built on a modular system of processes. Each process provides specific functionality for the system and runs in a protected memory space to ensure that problems with one process cannot impact the entire system. Multiple instances of a process can run on a single node, and multiple threads of execution can run on each process instance.

Under normal operating conditions, processes are managed automatically by the Cisco IOS XR software. Processes are started, stopped, or restarted as required by the running configuration of the router. In addition, processes are checkpointed to optimize performance during process restart and automatic switchover.

What Is Process Placement?

Process placement is the assignment of placeable processes to specific locations, such as an RP or DRP installed in the router. Process placement is configured and managed for each Secure Domain Router (SDR) in the system.

Placeable processes include all routing processes, such as Open Shortest Path First Protocol (OSPF), Border Gateway Protocol (BGP), and multicast routing.

Default Placement Policy

In a new system, processes are distributed according to their affinity values among the available RP and DRP nodes and node pairs in a SDR.

Note The default process policy that is shipped on the system upon startup is suitable for general purposes. While customizing is possible, there is no requirement to change the process placement. If you believe the a change is required, you should work closely with Cisco personnel to ensure that the impact to your system is contained to just an instance of a process to avoid any undesirable results.

Following is the default placement policy:

- Processes have a preference to run on paired nodes (nodes that have an associated standby node).
- Processes have a preference to remain on their current node. Therefore, processes do not move automatically, unless the unpaired node (or both nodes in a node pair) on which they are running fails. If the node fails, and there is no standby node, the processes are restarted on a different node.
- When a new node pair is added, the following rules apply:
  - The currently running processes are not automatically moved to the new cards.
  - The general preference is for new processes (such as a new ISIS instance) to start on the new node pair, which contains the most available CPU and memory resources in the system.

- Other affinity settings may override the general preference. For example, if the IS-IS process has a strong affinity to run on the same node where ipv4_io is running, then IS-IS would be started on that node, and not the new node-pair.

Reasons to Change the Default Process Placement

Although the default process policy that is shipped on the system upon startup is suitable for general purposes, changes to the router configuration can result in the need for processes to be rebalanced among the available CPU and memory resources.

When a system is initially booted, the system assumes that all processes use the same amount of memory, thereby treating each process as equivalent. As the configuration grows, however, the CPU load and memory requirements of some application processes increase. Centralized applications may need a larger portion of the RP and DRP resources, or distributed applications may require additional instances of processes to be started on new DRPs.

In addition, when a new RP or DRP is added to a system, only new processes or process instances are added to the node. This could result in some processes with too few resources, while the newer RP and DRP cards are underutilized.

Therefore, as the software configuration changes, or hardware is added, it may become necessary to rebalance processes among the available RPs and DRPs in an SDR .

Reoptimizing Process Placements

The easiest and most reliable method for users to redistribute processes among the available RPs and DRPs in an SDR is with the **placement reoptimize** command.

During router operation, the actual resource usage of each process is collected and compared to the router configuration and network topology. An ideal configuration for process placement is created and updated in real time.

To implement this ideal process placement configuration, enter the **placement reoptimize** command in EXEC mode. Before the changes are made, the system displays a summary of the predicted changes. You can either accept the changes or cancel the operation.

See Reoptimizing Process Placement, on page 303 for detailed instructions.

Reconfiguring Process Placements

You can also change the process placement *affinities*, or preferences, to override the default policies. For example, you may learn that some processes perform better on the primary node pair of the SDR, or that some processes have better high-availability characteristics when running on a paired node (a node with a standby partner). Other processes might benefit from co-location or by being assigned to nodes far apart from each other.

Note Consult with your technical support representative before changing the default process placement configuration. Incorrect configurations can cause system error, poor performance or downtime.

Recommended Guidelines for Process Placement

The following are a few recommended guidelines for changes to the process placement configuration:

- Generally, the process placement feature functions well upon system startup; fine tuning is seldom required.
- Use the EXEC mode command **placement reoptimize** , as described in the Reoptimizing Process Placements , on page 301 to automatically redistribute the processes among the available RPs and DRPs.
- Keep process placement policy changes to a minimum, and always consult technical support personnel before implementation.

Process Placement Based on Memory Consumption

You can change process placements based on memory use of processes in an SDR. Memory use is expressed in terms of the memory "footprint of the placeable process. The system attempts to spread the load among the nodes without exceeding their memory capacity. In addition, the system computes the affinity values to determine the best placement.

Cisco IOS XR software assumes that every placeable process uses one megabyte of memory.

For detailed instructions, see Setting Memory Consumption Thresholds, on page 304.

Changing Process Affinities

Process placement can also be controlled by changing the *affinities*, or preferences, of a process or process group. The following types of process affinities are operator configurable:

- affinity location set
- affinity location type
- affinity program
- affinity self

affinity location set

This affinity specifies a preference for a process to run on a specific node pair or set of node pairs. A node pair is either an active and standby pair of nodes [hosted on route processors (RPs) or distributed RPs], or a single active node on an RP or DRP that does not have a standby.

By containing references to specific location identifiers, location set affinities are more specific to the individual SDR in which they are used than other affinity types.

affinity location type

This affinity specifies a preference for a process to run on a particular location type. Available location types are as follows:

- **paired**—RP nodes that have an associated standby node
- **primary**—Primary RP node for the SDR (also known as the DSDRSC)
- **current** —Current node. A process's affinity to its current node characterizes its preference to remain on the same node where possible.

You configure the placement policy to allow certain processes to stay where they are (**current**) or move by specifying the various affinity values. The higher the positive value of an affinity, the stronger the requirement that the process run at a location, and so on. A low or zero point value indicates a weaker requirement (or no preference) that a process run at a location.

affinity program

This affinity specifies a preference for a process to run on the same node as another process, or to run on a different node than another process. You would want to use this affinity in the case that certain processes perform better when they are running together on the same node (**attract**); or on different nodes, apart from each other (**repulse**).

affinity self

This affinity adjusts placement decisions when multiple instances of a process are started. An attract (positive) affinity indicates a preference to have all instances of a process run on the same node, while a repulse (negative) affinity indicates a preference to have each instance of a process run on different nodes.

Hierarchical Placement Policy

When you configure placement policies, you must remember that affinities are applied to the software in a hierarchical way.

Affinities applied to process instances take precedence over affinities applied to a process class. In the following example, all OSPF instances have a preference to run on the primary RP of the SDR, but only OSPF instance 10 has a preference to run on a paired node:

```
RP/0/RP0/CPU0:router(config)# placement program ospf
RP/0/RP0/CPU0:router(config-place)# affinity location-type primary attract 200

RP/0/RP0/CPU0:router(config)# placement program ospf instance 10
RP/0/RP0/CPU0:router(config-place)# affinity location-type paired attract 200
```

Class affinities take precedence over default process affinities. In the following example, all OSPF instances have a preference to be placed on unpaired nodes. This overrides the default policy for all processes to prefer paired nodes.

```
RP/0/RP0/CPU0:router(config)# placement program ospf
RP/0/RP0/CPU0:router(config-place)# affinity location-type paired repulse 200
```

How to Configure Cisco IOS XR Process Placement

Reoptimizing Process Placement

This task reoptimizes the placeable processes among the available RP and DRP nodes according to memory and CPU usage.

SUMMARY STEPS

1. **placement reoptimize**

2. Use one of the following commands:
   - **yes**
   - **no**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**placement reoptimize** **Example:** `RP/0/RP0/CPU0:router# placement reoptimize`	Displays the predicted changes of the optimization.
Step 2	Use one of the following commands: • **yes** • **no** **Example:** `RP/0/RP0/CPU0:router# yes`	Accepts or rejects the changes.

Setting Memory Consumption Thresholds

SUMMARY STEPS

1. **show placement policy global**
2. **configure**
3. **placement memory** {**maximum** | **threshold**} *value*
4. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show placement policy global** **Example:** `RP/0/RP0/CPU0:router# show placement policy global`	Displays the current memory settings.
Step 2	**configure**	
Step 3	**placement memory** {**maximum**	**threshold**} *value* **Example:** `RP/0/RP0/CPU0:router(config)# placement memory maximum 80`	Use **maximum** *value* keyword and argument to set the maximum percentage of memory that can be used on a node (based on the estimated memory usage of the processes). Use the **threshold** *value* keyword and argument to define the memory load level to trigger migration. The system

	Command or Action	Purpose
		attempts to balance all nodes at or below the threshold memory percentage. In other words, the system does not place a process on a node that has exceeded the threshold value, unless all other nodes have also reached their thresholds (or unless some other large affinity overrides this consideration).

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-place)# end`
	or
	`RP/0/RP0/CPU0:router(config-place)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?[cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Creating a Location Set Affinity

This task sets the affinity of a placement program (process) to or from node pairs.

SUMMARY STEPS

1. **configure**
2. **placement program** {*program* [**instance** *instance*] | **default**}
3. **affinity location-set** *node-id1* [*node-id2*] {**attract** *strength* | **repulse** *strength* | **default** | **none**}
4. Use one of the following commands:
   - **end**
   - **commit**
5. **show placement location** {*node-id* | **all**}
6. **show placement program** {*program* | **all**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**placement program** {*program* [**instance** *instance*] | **default**} **Example:** `RP/0/RP0/CPU0:router(config)# placement program ospf`	Enters placement program configuration mode.
Step 3	**affinity location-set** *node-id1* [*node-id2*] {**attract** *strength* | **repulse** *strength* | **default** | **none**} **Example:** `RP/0/RP0/CPU0:router(config-place)# affinity location-set 0/1/cpu0 0/1/cpu1 attract 200`	Sets the affinity of a placement program (process) to or from node pairs. To specify multiple nodes, enter the value of the *node-id* argument for each node. You can specify up to 5 nodes.

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-place)# end`
	or
	`RP/0/RP0/CPU0:router(config-place)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?[cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Step 5	**show placement location** {*node-id* | **all**} **Example:** `RP/0/RP0/CPU0:router# show placement location all`	Displays the location of a placement process.
Step 6	**show placement program** {*program* | **all**} **Example:** `RP/0/RP0/CPU0:router# show placement program ospf`	Displays the operational state for each placement program.

Creating a Location Type Affinity

This task sets affinity of a placement program (process) to or from a location type.

SUMMARY STEPS

1. **configure**
2. **placement program** {*program* [**instance** *instance*] | **default**}
3. **affinity location-type** {**current** | **paired** | **primary**} {**attract** *strength* | **repulse** *strength* | **default** | **none**}
4. Use one of the following commands:
   - **end**
   - **commit**
5. **show placement location** {*node-id* | **all**}
6. **show placement program** {*program* | **all**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**placement program** {*program* [**instance** *instance*]	**default**} **Example:** `RP/0/RP0/CPU0:router(config)# placement program bgp`	Enters placement program configuration mode.
Step 3	**affinity location-type** {**current**	**paired**	**primary**} {**attract** *strength*	**repulse** *strength*	**default**	**none**} **Example:** `RP/0/RP0/CPU0:router(config-place)# affinity location-type current attract 10`	Sets the affinity of a placement program (process) to or from a location type. • This example shows how to place Border Gateway Protocol (BGP) in the most optimal location at run time when load balancing is required. BGP will not be tied to a node pair but move when necessary.

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-place)# end`
	or
	`RP/0/RP0/CPU0:router(config-place)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?[cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.

	Command or Action	Purpose
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes. • Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.
Step 5	**show placement location** {*node-id* | **all**} **Example:** `RP/0/RP0/CPU0:router# show placement location all`	Displays the location of a placement process.
Step 6	**show placement program** {*program* | **all**} **Example:** `RP/0/RP0/CPU0:router# show placement program bgp`	Displays the operational state for each placement program.

Creating a Program Affinity

This task sets the affinity of a placement program (process) to or from another program.

SUMMARY STEPS

1. **configure**
2. **placement program** {*program* [**instance** *instance*] | **default**}
3. **affinity program** *program* {**attract** *strength* | **repulse** *strength* | **default** | **none**}
4. Use one of the following commands:
   - **end**
   - **commit**
5. **show placement location** {*node-id* | **all**}
6. **show placement program** {*program* | **all**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**placement program** {*program* [**instance** *instance*]	**default**} **Example:** `RP/0/RP0/CPU0:router(config)# placement program ipv4_rib`	Enters placement program configuration mode.
Step 3	**affinity program** *program* {**attract** *strength*	**repulse** *strength*	**default**	**none**}	Sets the affinity of a placement program (process) to or from another program.

	Command or Action	Purpose
	Example: `RP/0/RP0/CPU0:router(config-place)# affinity program ipv6_rib repulse 200`	• This example shows how to keep IPv4 and IPv6 Routing Information Bases (RIBs) apart.

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-place)# end`
	or
	`RP/0/RP0/CPU0:router(config-place)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?[cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Step 5	**show placement location** {*node-id* | **all**} **Example:** `RP/0/RP0/CPU0:router# show placement location all`	Displays the location of a placement process.
Step 6	**show placement program** {*program* | **all**} **Example:** `RP/0/RP0/CPU0:router# show placement program all`	Displays the operational state for each placement program.

Creating a Self Affinity

This task sets the affinity of a placement program (process) to or from one of its own instances.

SUMMARY STEPS

1. **configure**
2. **placement program** *program* {**instance** *instance* | **default**}
3. **affinity self** {**attract** *strength*| **repulse** *strength* | **default** | **none**}
4. Use one of the following commands:
   - **end**
   - **commit**

5. **show placement location** {*node-id* | **all**}
6. **show placement program** {*program* | **all**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**placement program** *program* {**instance** *instance* | **default**} **Example:** `RP/0/RP0/CPU0:router(config)# placement program bgp`	Enters placement program configuration mode.
Step 3	**affinity self** {**attract** *strength*| **repulse** *strength* | **default** | **none**} **Example:** `RP/0/RP0/CPU0:router(config-place)# affinity self repulse 200`	Sets the affinity of a placement program (process) to or from one of its own instances.

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-place)# end`
	or
	`RP/0/RP0/CPU0:router(config-place)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting (yes/no/cancel)?[cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Step 5	**show placement location** {*node-id* | **all**} **Example:** `RP/0/RP0/CPU0:router# show placement location all`	Displays the location of a placement process.
Step 6	**show placement program** {*program* | **all**} **Example:** `RP/0/RP0/CPU0:router# show placement program bgp`	Displays the operational state for each placement program.

Configuration Examples for Process Placement

This section contains examples to view the processes that are placeable in an SDR.

If you believe that a custom reconfiguration of the processes on your system is required, you should work closely with Cisco personnel to ensure that the impact to your system is contained to just an instance of a process to avoid any undesirable results.

To learn the processes that are placeable, enter the **show placement program all** command in EXEC mode.

```
RP/0/RP0/CPU0:router# show placement program all

  Mon Aug 18 17:13:15.155 PST DST

  If a program is shown as having 'rejected locations' (i.e., locations on which
  it cannot be placed), the locations in question can been seen using the "show
  placement policy program" command.

  If a program has been placed but not yet started, the amount of time elapsed
  since the program was placed is shown in the 'waiting to start' field.

  Parentheses around the node indicate that the node has not yet fully booted.
    This will be true of standby nodes.

  Program                        Placed at location            # rejected  Waiting
                                                               locations   to start
  ------------------------------------------------------------------------------
  li_mgr                         0/RP0/CPU0 (0/RP1/CPU0)
  rsi_master                     0/RP0/CPU0 (0/RP1/CPU0)
  statsd_manager                 0/RP0/CPU0 (0/RP1/CPU0)
  ipv4_rib                       0/RP0/CPU0 (0/RP1/CPU0)
  ipv6_rib                       0/RP0/CPU0 (0/RP1/CPU0)
  policy_repository              0/RP0/CPU0 (0/RP1/CPU0)
  ipv4_mpa                       0/RP0/CPU0 (0/RP1/CPU0)
  ipv6_mpa                       0/RP0/CPU0 (0/RP1/CPU0)
  bfd                            0/RP0/CPU0 (0/RP1/CPU0)
  domain_services                0/RP0/CPU0 (0/RP1/CPU0)
  ftp_fs                         0/RP0/CPU0 (0/RP1/CPU0)
  rcp_fs                         0/RP0/CPU0 (0/RP1/CPU0)
  tftp_fs                        0/RP0/CPU0 (0/RP1/CPU0)
  ipv4_connected                 0/RP0/CPU0 (0/RP1/CPU0)
  ipv4_local                     0/RP0/CPU0 (0/RP1/CPU0)
  ipv4_rump                      0/RP0/CPU0 (0/RP1/CPU0)
  ipv6_connected                 0/RP0/CPU0 (0/RP1/CPU0)
  ipv6_local                     0/RP0/CPU0 (0/RP1/CPU0)
  ipv6_rump                      0/RP0/CPU0 (0/RP1/CPU0)
  atmgcmgr                       0/RP0/CPU0 (0/RP1/CPU0)
  eem_metric_dir                 0/RP0/CPU0 (0/RP1/CPU0)
  l2tp_mgr                       0/RP0/CPU0 (0/RP1/CPU0)
  l2vpn_mgr                      0/RP0/CPU0 (0/RP1/CPU0)
  rt_check_mgr                   0/RP0/CPU0 (0/RP1/CPU0)
  ipv4_static                    0/RP0/CPU0 (0/RP1/CPU0)
  isis instance lab              0/RP0/CPU0 (0/RP1/CPU0)
  ospf instance 100              0/RP0/CPU0 (0/RP1/CPU0)
  isis_uv                        0/RP0/CPU0 (0/RP1/CPU0)
  ospf_uv                        0/RP0/CPU0 (0/RP1/CPU0)
  mpls_vpn_mib                   0/RP0/CPU0 (0/RP1/CPU0)
  rsvp                           0/RP0/CPU0 (0/RP1/CPU0)
  mpls_ldp                       0/RP0/CPU0 (0/RP1/CPU0)
  lspv_server                    0/RP0/CPU0 (0/RP1/CPU0)
  ospf instance 0                0/RP0/CPU0 (0/RP1/CPU0)
```

```
ospfv3 instance 0             0/RP0/CPU0 (0/RP1/CPU0)
ospfv3_uv                     0/RP0/CPU0 (0/RP1/CPU0)
```

Additional References

The following sections provide references related to Cisco IOS XR Process Placement.

Related Documents

Related Topic	Document Title
Cisco IOS XR process placement commands	*Process and Memory Management Commands on Cisco IOS XR software* module of *System Management Command Reference for isco CRS Routers*
Cisco IOS XR master command index	*Cisco IOS XR Commands Master List for the Cisco CRS Router*
Getting started with Cisco IOS XR software	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*
Information about user groups and task IDs	*Configuring AAA Services on Cisco IOS XR software* module of *System Security Configuration Guide for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

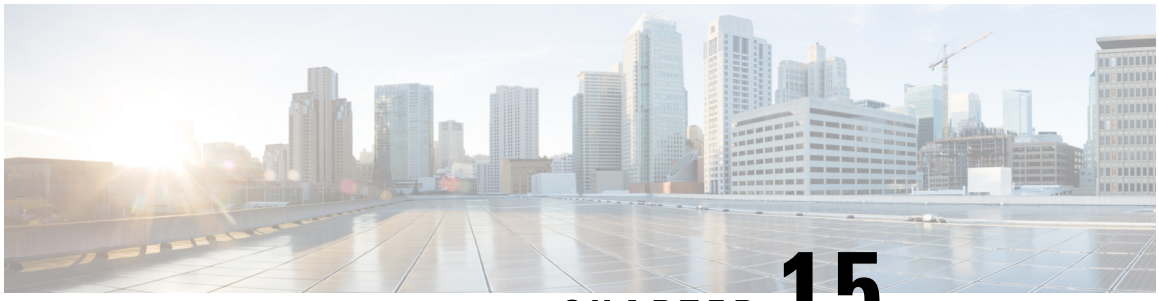
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 15

Configuring Physical and Virtual Terminals

Line templates define standard attribute settings for incoming and outgoing transport over physical and virtual terminal lines (vtys). Vty pools are used to apply template settings to ranges of vtys.

Note Before creating or modifying the vty pools, enable the telnet server using the **telnet server** command in Global Configuration mode. See *IP Addresses and Services Configuration Guide for Cisco CRS Routers* and *IP Addresses and Services Command Reference for Cisco CRS Routers* for more information.

This module describes the new and revised tasks you need to implement physical and virtual terminals on your Cisco IOS XR network.

For more information about physical and virtual terminals on the Cisco IOS XR software and complete descriptions of the terminal services commands listed in this module, see Related Documents, on page 324. To locate documentation for other commands that might appear in the course of running a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 34: Feature History for Implementing Physical and Virtual Templates on Cisco IOS XR Software

Release	Modification
Release 2.0	This feature was introduced.
Release 3.8.0	The keyword **fm** in the **vty-pool** command was changed to **eem** .

This module contains the following topics:

Prerequisites for Implementing Physical and Virtual Terminals

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Information About Implementing Physical and Virtual Terminals

To implement physical and virtual terminals, you need to understand the concepts in this section.

Line Templates

The following line templates are available in the Cisco IOS XR software.

- Default line template—The default line template that applies to a physical and virtual terminal lines.
- Console line template—The line template that applies to the console line.
- User-defined line templates—User-defined line templates that can be applied to a range of virtual terminal lines.

Line Template Configuration Mode

Changes to line template attributes are made in line template configuration mode. To enter line template configuration mode, issue the **line** command from Global Configuration mode, specifying the template to be modified. These line templates can be configured with the **line** command:

- console—console template
- default—default template
- template—user-defined template

After you specify a template with the **line** command, the router enters line template configuration mode where you can set the terminal attributes for the specified line. This example shows how to specify the attributes for the console:

```
RP/0/RP0/CPU0:router(config)# line console
RP/0/RP0/CPU0:router(config-line)#
```

From line template configuration mode, use the online help feature (**?**) to view all available options. Some useful options include:

- absolute-timeout—Specifies a timeout value for line disconnection.
- escape-character—Changes the line escape character.
- exec-timeout—Specifies the EXEC timeout.
- length—Sets the number of lines displayed on the screen.

- session-limit—Specifies the allowable number of outgoing connections.
- session-timeout—Specifies an interval for closing the connection if there is no input traffic.
- timestamp—Displays the timestamp before each command.
- width—Specifies the width of the display terminal.

Line Template Guidelines

The following guidelines apply to modifying the console template and to configuring a user-defined template:

- Modify the templates for the physical terminal lines on the router (the console port) from line template configuration mode. Use the **line console** command from Global Configuration mode to enter line template configuration mode for the console template.
- Modify the template for virtual lines by configuring a user-defined template with the **line** *template-name* command, configuring the terminal attributes for the user-defined template from line template configuration, and applying the template to a range of virtual terminal lines using the **vty pool** command.

Attributes not defined in the console template, or any virtual template, are taken from the default template.

The default settings for the default template are described for all commands in line template configuration mode in the *Terminal Services Commands on the Cisco IOS XR Software* module in *System Management Command Reference for isco CRS Routers*.

Note Before creating or modifying the vty pools, enable the telnet server using the **telnet server** command in Global Configuration mode. See *IP Addresses and Services Configuration Guide for Cisco CRS Routers* and *IP Addresses and Services Command Reference for Cisco CRS Routers* for more information.

Terminal Identification

The physical terminal lines for the console port is identified by its location, expressed in the format of *rack*/*slot*/*module* , on the active or standby route processor (RP) where the respective console port resides. For virtual terminals, physical location is not applicable; the Cisco IOS XR software assigns a vty identifier to vtys according to the order in which the vty connection has been established.

vty Pools

Each virtual line is a member of a pool of connections using a common line template configuration. Multiple vty pools may exist, each containing a defined number of vtys as configured in the vty pool. The Cisco IOS XR software supports the following vty pools by default:

- Default vty pool—The default vty pool consists of five vtys (vtys 0 through 4) that each reference the default line template.
- Default fault manager pool—The default fault manager pool consists of six vtys (vtys 100 through 105) that each reference the default line template.

In addition to the default vty pool and default fault manager pool, you can also configure a user-defined vty pool that can reference the default template or a user-defined template.

When configuring vty pools, follow these guidelines:

- The vty range for the default vty pool must start at vty 0 and must contain a minimum of five vtys.
- The vty range from 0 through 99 can reference the default vty pool.
- The vty range from 5 through 99 can reference a user-defined vty pool.
- The vty range from 100 is reserved for the fault manager vty pool.
- The vty range for fault manager vty pools must start at vty 100 and must contain a minimum of six vtys.
- A vty can be a member of only one vty pool. A vty pool configuration will fail if the vty pool includes a vty that is already in another pool.
- If you attempt to remove an active vty from the active vty pool when configuring a vty pool, the configuration for that vty pool will fail.

How to Implement Physical and Virtual Terminals on Cisco IOS XR Software

Modifying Templates

This task explains how to modify the terminal attributes for the console and default line templates. The terminal attributes that you set will modify the template settings for the specified template.

SUMMARY STEPS

1. **configure**
2. **line** {**console** | **default**}
3. Configure the terminal attribute settings for the specified template using the commands in line template configuration mode.
4. Use one of the following commands:
   - **end**
   - **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

Step 2	**line** {**console** | **default**}
	Example:
	`RP/0/RP0/CPU0:router(config)# line console`
	or
	`RP/0/RP0/CPU0:router(config)# line default`
		Enters line template configuration mode for the specified line template.
		• **console** —Enters line template configuration mode for the console template.
		• **default** —Enters line template configuration mode for the default line template.

	Command or Action	Purpose
Step 3	Configure the terminal attribute settings for the specified template using the commands in line template configuration mode.	—

Step 4	Use one of the following commands:
	• **end**
	• **commit**
	Example:
	`RP/0/RP0/CPU0:router(config-line)# end`
	or
	`RP/0/RP0/CPU0:router(config-line)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:`
		• Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		• Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Entering **cancel** leaves the router in the current configuration session without exiting or committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Creating and Modifying vty Pools

This task explains how to create and modify vty pools.

You can omit Step 3, on page 320 to Step 5, on page 320 if you are configuring the default line template to reference a vty pool.

SUMMARY STEPS

1. **configure**
2. **telnet** {**ipv4** | **ipv6**} **server max-servers** *limit*
3. **line template** *template-name*
4. Configure the terminal attribute settings for the specified line template using the commands in line template configuration mode.
5. **exit**
6. **vty-pool** {**default** | *pool-name* | **eem**} *first-vty last-vty* [**line-template** {**default** | *template-name*}]
7. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**telnet** {**ipv4** | **ipv6**} **server max-servers** *limit* **Example:** `RP/0/RP0/CPU0:router(config)# telnet ipv4 server max-servers 10`	Specifies the number of allowable Telnet servers. Up to 100 Telnet servers are allowed. **Note** By default no Telnet servers are allowed. You must configure this command in order to enable the use of Telnet servers.
Step 3	**line template** *template-name* **Example:** `RP/0/RP0/CPU0:router(config)# line template 1`	Enters line template configuration mode for a user-defined template.
Step 4	Configure the terminal attribute settings for the specified line template using the commands in line template configuration mode.	—
Step 5	**exit** **Example:** `RP/0/RP0/CPU0:router(config-line)# exit`	Exits line template configuration mode and returns the router to global configuration mode.

Step 6	**vty-pool** {**default** | *pool-name* | **eem**} *first-vty last-vty* [**line-template** {**default** | *template-name*}]
	Example:
	`RP/0/RP0/CPU0:router(config)# vty-pool default 0 5 line-template default`
	or
	`RP/0/RP0/CPU0:router(config)# vty-pool pool1 5 50 line-template template1`
	or
	`RP/0/RP0/CPU0:router(config)# vty-pool eem 100 105 line-template template1`
		Creates or modifies vty pools.
		• If you do not specify a line template with the **line-template** keyword, a vty pool defaults to the default line template.
		• **default** —Configures the default vty pool.
		  • The default vty pool must start at vty 0 and must contain a minimum of five vtys (vtys 0 through 4).
		  • You can resize the default vty pool by increasing the range of vtys that compose the default vty pool.
		• *pool-name* —Creates a user-defined vty pool.
		  • A user-defined pool must start at least at vty 5, depending on whether the default vty pool has been resized.
		  • If the range of vtys for the default vty pool has been resized, use the first range value free from the default line template. For example, if the range of vtys for the default vty pool has been configured to include 10 vtys (vty 0 through 9),

	Command or Action	Purpose
		the range value for the user-defined vty pool must start with vty 10. • **eem** —Configures the embedded event manager pool. • The default embedded event manager vty pool must start at vty 100 and must contain a minimum of six vtys (vtys 100 through 105). • **line-template** *template-name* —Configures the vty pool to reference a user-defined template.
Step 7	**commit**	

Monitoring Terminals and Terminal Sessions

This task explains how to monitor terminals and terminal sessions using the **show** EXEC commands available for physical and terminal lines.

Note The commands can be entered in any order.

SUMMARY STEPS

1. (Optional) **show line** [**aux location** *node-id* | **console location** *node-id* | **vty** *number*]
2. (Optional) **show terminal**
3. (Optional) **show users**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **show line** [**aux location** *node-id* | **console location** *node-id* | **vty** *number*] **Example:** `RP/0/RP0/CPU0:router# show line`	Displays the terminal parameters of terminal lines. • Specifying the **show line aux location** *node-id* EXEC command displays the terminal parameters of the auxiliary line. • Specifying the **show line console location** *node-id* EXEC command displays the terminal parameters of the console. • For the **location** *node-id* keyword and argument, enter the location of the Route Processor (RP) on which the respective auxiliary or console port resides. • The *node-id* argument is expressed in the format of *rack*/*slot*/*module* .

	Command or Action	Purpose
		• Specifying the **show line vty** *number* EXEC command displays the terminal parameters for the specified vty.
Step 2	(Optional) **show terminal** **Example:** `RP/0/RP0/CPU0:router# show terminal`	Displays the terminal attribute settings for the current terminal line.
Step 3	(Optional) **show users** **Example:** `RP/0/RP0/CPU0:router# show users`	Displays information about the active lines on the router.

Craft Panel Interface

The Craft Panel is an easily-accessible and user-friendly interface which assists the field operator in troubleshooting the router. It consists of a LCD display and three LEDs. The LEDs indicate minor, major and critical alarms.

For more details of the Craft Panel Interface, refer the *Hardware and System set-up guides.*

Configuration Examples for Implementing Physical and Virtual Terminals

Modifying the Console Template: Example

This configuration example shows how to modify the terminal attribute settings for the console line template:

```
line console
  exec-timeout 0 0
  escape-character 0x5a
  session-limit 10
  disconnect-character 0x59
  session-timeout 100
  transport input telnet
  transport output telnet
```

In this configuration example, the following terminal attributes are applied to the console line template:

- The EXEC time out for terminal sessions is set to 0 minutes, 0 seconds. Setting the EXEC timeout to 0 minutes and 0 seconds disables the EXEC timeout function; thus, the EXEC session for the terminal session will never time out.

- The escape character is set to the 0x5a hexadecimal value (the 0x5a hexadecimal value translates into the “Z” character).
- The session limit for outgoing terminal sessions is set to 10 connections.
- The disconnect character is set to 0x59 hexadecimal value (the 0x59 hexadecimal character translates into the “Y” character).
- The session time out for outgoing terminal sessions is set to 100 minutes (1 hour and 40 minutes).
- The allowed transport protocol for incoming terminal sessions is Telnet.
- The allowed transport protocol for outgoing terminal sessions is Telnet.

To verify that the terminal attributes for the console line template have been applied to the console, use the **show line** command:

```
RP/0/RP0/CPU0:router# show line console location 0/0/CPU0

Tty             Speed     Modem  Uses   Noise Overruns           Acc I/O
*  con0/0/CPU0      9600        -     -       -      0/0               -/-

Line con0_0_CPU0, Location "Unknown", Type "Unknown"
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600, 1 parity, 2 stopbits, 8 databits
Template: console
Config:
Allowed transports are telnet.
```

Modifying the Default Template: Example

This configuration example shows how to override the terminal settings for the default line template:

```
line default
  exec-timeout 0 0
  width 512
  length 512
```

In this example, the following terminal attributes override the default line template default terminal attribute settings:

- The EXEC timeout for terminal sessions is set to 0 minutes and 0 seconds. Setting the EXEC timeout to 0 minutes and 0 seconds disables the EXEC timeout function; thus, the EXEC session for the terminal session will never time out (the default EXEC timeout for the default line template is 10 minutes).
- The width of the terminal screen for the terminals referencing the default template is set to 512 characters (the default width for the default line template is 80 characters).
- The length, the number of lines that will display at one time on the terminal referencing the default template, is set to 512 lines (the default length for the default line template is 24 lines).

Configuring a User-Defined Template to Reference the Default vty Pool: Example

This configuration example shows how to configure a user-defined line template (named test in this example) for vtys and to configure the line template test to reference the default vty pool:

```
line template test
  exec-timeout 100 0
```

```
  width 100
  length 100
  exit
vty-pool default 0 4 line-template test
```

Configuring a User-Defined Template to Reference a User-Defined vty Pool: Example

This configuration example shows how to configure a user-defined line template (named test2 in this example) for vtys and to configure the line template test to reference a user-defined vty pool (named pool1 in this example):

```
line template test2
  exec-timeout 0 0
  session-limit 10
  session-timeout 100
  transport input all
  transport output all
  exit
vty-pool pool1 5 50 line-template test2
```

Configuring a User-Defined Template to Reference the Fault Manager vty Pool: Example

This configuration example shows how to configure a user-defined line template (named test3 in this example) for vtys and to configure the line template test to reference the fault manager vty pool:

```
line template test3
  width 110
  length 100
  session-timeout 100
  exit
  vty-pool eem 100 106 line-template test3
```

Additional References

The following sections provide references related to implementing physical and virtual terminals on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Cisco IOS XR terminal services commands	*Terminal Services Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*
Cisco IOS XR command master index	*Cisco IOS XR Commands Master List for the Cisco CRS Router*
Information about getting started with Cisco IOS XR software	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*

Related Topic	Document Title
Information about user groups and task IDs	*Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

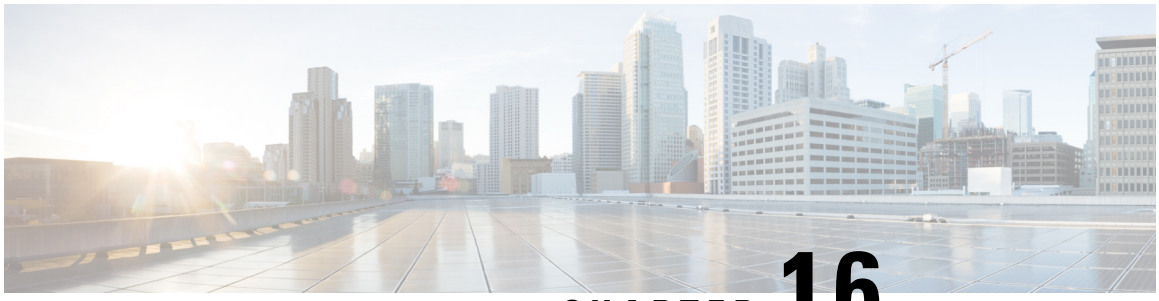
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 16

Configuring Simple Network Management Protocol

Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

This module describes the new and revised tasks you need to implement SNMP on your Cisco IOS XR network.

For detailed conceptual information about SNMP on the Cisco IOS XR software and complete descriptions of the SNMP commands listed in this module, see Related Documents, on page 350. For information on specific MIBs, refer to *Cisco CRS and Cisco XR 12000 Series Router MIB Overview*. To locate documentation for other commands that might appear in the course of performing a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 35: Feature History for Implementing SNMP on Cisco IOS XR Software

Release	Modification
Release 2.0	This feature was introduced.
Release 3.6.0	Support was added for encrypted community strings. Support was added for setting IPv4 precedence and DSCP values. The **clear snmp counters** command was added.
Release 3.8.0	Support was added for SNMP contexts.
Release 3.9.0	Support was added for packet loss monitoring. Support was added for 3DES and AES encryption. The ability to preserve ENTITY-MIB and CISCO-CLASS-BASED-QOS-MIB data was added.
Release 4.2.0	Support was added for SNMP over IPv6.

This module contains the following topics:

Prerequisites for Implementing SNMP

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Restrictions for SNMP Use on Cisco IOS XR Software

SNMP outputs are only 32-bits wide and therefore cannot display any information greater than 2^{32}. 2^{32} is equal to 4.29 Gigabits. Note that a 10 Gigabit interface is greater than this and so if you are trying to display speed information regarding the interface, you might see concatenated results.

Information About Implementing SNMP

To implement SNMP, you need to understand the concepts described in this section.

SNMP Functional Overview

The SNMP framework consists of three parts:

- SNMP manager
- SNMP agent
- Management Information Base (MIB)

SNMP Manager

The SNMP manager is the system used to control and monitor the activities of network hosts using SNMP. The most common managing system is called a *network management system* (NMS). The term NMS can be applied to either a dedicated device used for network management, or the applications used on such a device. A variety of network management applications are available for use with SNMP. These features range from simple command-line applications to feature-rich graphical user interfaces (such as the CiscoWorks 2000 line of products).

SNMP Agent

The SNMP agent is the software component within the managed device that maintains the data for the device and reports these data, as needed, to managing systems. The agent and MIB reside on the router. To enable the SNMP agent, you must define the relationship between the manager and the agent.

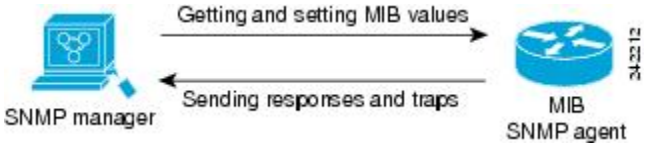
MIB

The *Management Information Base* (MIB) is a virtual information storage area for network management information, which consists of collections of managed objects. Within the MIB there are collections of related objects, defined in MIB modules. MIB modules are written in the SNMP MIB module language, as defined in STD 58, RFC 2578, RFC 2579, and RFC 2580. Note that individual MIB modules are also referred to as MIBs; for example, the Interfaces Group MIB (IF-MIB) is a MIB module within the MIB on your system.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change through Get or Set operations. A manager can get a value from an agent or store a value into that agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to manager requests to get or set data.

This figure illustrates the communications relationship between the SNMP manager and agent. A manager can send the agent requests to get and set MIB values. The agent can respond to these requests. Independent of this interaction, the agent can send unsolicited notifications (traps) to the manager to notify the manager of network conditions.

Figure 5: Communication Between an SNMP Agent and Manager



Related Topics

Additional References, on page 350

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. On Cisco IOS XR software, unsolicited (asynchronous) notifications can be generated only as *traps*. Traps are messages alerting the SNMP manager to a condition on the network. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

Note Inform requests (inform operations) are supported in Cisco IOS XR software from release 4.1 onwards. For more information see, http://www.cisco.com/c/en/us/td/docs/routers/asr9000/software/asr9k_r5-3/sysman/command/reference/b-sysman-cr53xasr/b-sysman-cr53xasr_chapter_010010.html#wp2863682680

Traps are less reliable than informs because the receiver does not send any acknowledgment when it receives a trap. The sender cannot determine if the trap was received. An SNMP manager that receives an inform request acknowledges the message with an SNMP response protocol data unit (PDU). If the manager does not receive an inform request, it does not send a response. If the sender never receives a response, the inform request can be sent again. Thus, informs are more likely to reach their intended destination.

However, traps are often preferred because informs consume more resources in the router and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request must be held in memory until a response is received or the request times out. Also, traps are sent only once, and an inform may be retried several times. The retries increase traffic and contribute to a higher overhead on the network. Thus, traps and inform requests provide a trade-off between reliability and resources.

Figure 6: Trap Received by the SNMP Manager

In this illustration, the agent router sends a trap to the SNMP manager. Although the manager receives the trap, it does not send any acknowledgment to the agent. The agent has no way of knowing that the trap reached its destination.

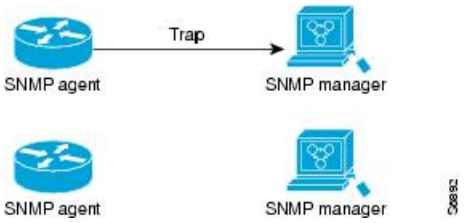
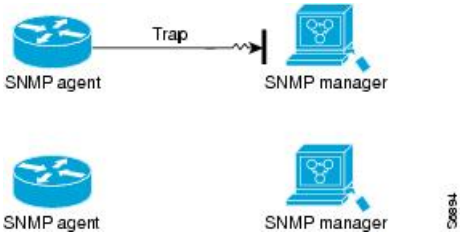


Figure 7: Trap Not Received by the SNMP Manager

In this illustration, the agent sends a trap to the manager, but the trap does not reach the manager. Because the agent has no way of knowing that the trap did not reach its destination, the trap is not sent again. The manager never receives the trap.



SNMP Versions

Cisco IOS XR software supports the following versions of SNMP:

- Simple Network Management Protocol Version 1 (SNMPv1)
- Simple Network Management Protocol Version 2c (SNMPv2c)
- Simple Network Management Protocol Version 3 (SNMPv3)

Both SNMPv1 and SNMPv2c use a community-based form of security. The community of managers able to access the agent MIB is defined by an IP address access control list and password.

SNMPv2c support includes a bulk retrieval mechanism and more detailed error message reporting to management stations. The bulk retrieval mechanism supports the retrieval of tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2c improved error handling support includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes now report the error type. Three kinds of exceptions are also reported: no such object exceptions, no such instance exceptions, and end of MIB view exceptions.

SNMPv3 is a security model. A *security model* is an authentication strategy that is set up for a user and the group in which the user resides. A *security level* is the permitted level of security within a security model. A combination of a security model and a security level will determine which security mechanism is employed when an SNMP packet is handled. See Table 37: SNMP Security Models and Levels, on page 332 for a list of security levels available in SNMPv3. The SNMPv3 feature supports RFCs 3411 to 3418.

You must configure the SNMP agent to use the version of SNMP supported by the management station. An agent can communicate with multiple managers; for this reason, you can configure the Cisco IOS-XR software to support communications with one management station using the SNMPv1 protocol, one using the SNMPv2c protocol, and another using SMNPv3.

Comparison of SNMPv1, v2c, and v3

SNMP v1, v2c, and v3 all support the following operations:

- get-request—Retrieves a value from a specific variable.
- get-next-request—Retrieves the value following the named variable; this operation is often used to retrieve variables from within a table. With this operation, an SNMP manager does not need to know the exact variable name. The SNMP manager searches sequentially to find the needed variable from within the MIB.
- get-response—Operation that replies to a get-request, get-next-request, and set-request sent by an NMS.
- set-request—Operation that stores a value in a specific variable.
- trap—Unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

The below table identifies other key SNMP features supported by the SNMP v1, v2c, and v3.

Table 36: SNMPv1, v2c, and v3 Feature Support

Feature	SNMP v1	SNMP v2c	SNMP v3
Get-Bulk Operation	No	Yes	Yes
Inform Operation	No	Yes (No on the Cisco IOS XR software)	Yes (No on the Cisco IOS XR software)
64 Bit Counter	No	Yes	Yes
Textual Conventions	No	Yes	Yes
Authentication	No	No	Yes
Privacy (Encryption)	No	No	Yes
Authorization and Access Controls (Views)	No	No	Yes

Security Models and Levels for SNMPv1, v2, v3

The security level determines if an SNMP message needs to be protected from disclosure and if the message needs to be authenticated. The various security levels that exist within a security model are as follows:

- noAuthNoPriv—Security level that does not provide authentication or encryption.
- authNoPriv—Security level that provides authentication but does not provide encryption.
- authPriv—Security level that provides both authentication and encryption.

Three security models are available: SNMPv1, SNMPv2c, and SNMPv3. The security model combined with the security level determine the security mechanism applied when the SNMP message is processed.

The below table identifies what the combinations of security models and levels mean.

Table 37: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	HMAC-MD5 or HMAC-SHA	No	Provides authentication based on the HMAC[2]-MD5[3] algorithm or the HMAC-SHA[4].
v3	authPriv	HMAC-MD5 or HMAC-SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES[5] 56-bit encryption in addition to authentication based on the CBC[6] DES (DES-56) standard.
v3	authPriv	HMAC-MD5 or HMAC-SHA	3DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides 168-bit 3DES[7] level of encryption.
v3	authPriv	HMAC-MD5 or HMAC-SHA	AES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides 128-bit AES[8] level of encryption.

2 Hash-Based Message Authentication Code
3 Message Digest 5
4 Secure Hash Algorithm
5 Data Encryption Standard
6 Cipher Block Chaining
7 Triple Data Encryption Standard
8 Advanced Encryption Standard

Use of 3DES and AES encryption standards requires that the security package (k9sec) be installed. For information on installing software packages, see *Upgrading and Managing Cisco IOS XR Software*.

SNMPv3 Benefits

SNMPv3 provides secure access to devices by providing authentication, encryption and access control. These added security benefits secure SNMP against the following security threats:

- Masquerade—The threat that an SNMP user may assume the identity of another SNMP user to perform management operations for which that SNMP user does not have authorization.
- Message stream modification—The threat that messages may be maliciously reordered, delayed, or replayed (to an extent that is greater than can occur through the natural operation of a subnetwork service) to cause SNMP to perform unauthorized management operations.
- Disclosure—The threat that exchanges between SNMP engines could be eavesdropped. Protecting against this threat may be required as a matter of local policy.

In addition, SNMPv3 provides access control over protocol operations on SNMP managed objects.

SNMPv3 Costs

SNMPv3 authentication and encryption contribute to a slight increase in the response time when SNMP operations on MIB objects are performed. This cost is far outweighed by the security advantages provided by SNMPv3.

This table shows the order of response time (from least to greatest) for the various security model and security level combinations.

Table 38: Order of Response Times from Least to Greatest

Security Model	Security Level
SNMPv2c	noAuthNoPriv
SNMPv3	noAuthNoPriv
SNMPv3	authNoPriv
SNMPv3	authPriv

User-Based Security Model

SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages have not been altered or destroyed in an unauthorized manner and that data sequences have not been altered to an extent greater than can occur nonmaliciously.
- Message origin authentication—Ensures that the claimed identity of the user on whose behalf received data was originated is confirmed.
- Message confidentiality—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMPv3 authorizes management operations only by configured users and encrypts SNMP messages.

USM uses two authentication protocols:

- HMAC-MD5-96 authentication protocol
- HMAC-SHA-96 authentication protocol

USM uses Cipher Block Chaining (CBC)-DES (DES-56) as the privacy protocol for message encryption.

View-Based Access Control Model

The View-Based Access Control Model (VACM) enables SNMP users to control access to SNMP managed objects by supplying read, write, or notify access to SNMP objects. It prevents access to objects restricted by views. These access policies can be set when user groups are configured with the **snmp-server group** command.

MIB Views

For security reasons, it is often valuable to be able to restrict the access rights of some groups to only a subset of the management information within the management domain. To provide this capability, access to a management object is controlled through MIB views, which contain the set of managed object types (and, optionally, the specific instances of object types) that can be viewed.

Access Policy

Access policy determines the access rights of a group. The three types of access rights are as follows:

- read-view access—The set of object instances authorized for the group when objects are read.
- write-view access—The set of object instances authorized for the group when objects are written.
- notify-view access—The set of object instances authorized for the group when objects are sent in a notification.

IP Precedence and DSCP Support for SNMP

SNMP IP Precedence and differentiated services code point (DSCP) support delivers QoS specifically for SNMP traffic. You can change the priority setting so that SNMP traffic generated in a router is assigned a specific QoS class. The IP Precedence or IP DSCP code point value is used to determine how packets are handled in weighted random early detection (WRED).

After the IP Precedence or DSCP is set for the SNMP traffic generated in a router, different QoS classes cannot be assigned to different types of SNMP traffic in that router.

The IP Precedence value is the first three bits in the type of service (ToS) byte of an IP header. The IP DSCP code point value is the first six bits of the differentiate services (DiffServ Field) byte. You can configure up to eight different IP Precedence markings or 64 different IP DSCP markings.

How to Implement SNMP on Cisco IOS XR Software

This section describes how to implement SNMP.

The **snmp-server** commands enable SNMP on Management Ethernet interfaces by default. For information on how to enable SNMP server support on other inband interfaces, see the *Implementing Management Plane Protection on Cisco IOS XR Software* module in *System Security Configuration Guide for Cisco CRS Routers*.

Configuring SNMPv3

This task explains how to configure SNMPv3 for network management and monitoring.

Note No specific command enables SNMPv3; the first **snmp-server** global configuration command (config), that you issue enables SNMPv3. Therefore, the sequence in which you issue the **snmp-server** commands for this task does not matter.

SUMMARY STEPS

1. **configure**

2. **snmp-server view** *view-name oid-tree* {**included** | **excluded**}
3. **snmp-server group** *name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *view*] [**write** *view*] [**notify** *view*] [*access-list-name*]
4. **snmp-server user** *username groupname* {**v1** | **v2c** | **v3** [**auth** {**md5** | **sha**} {**clear** | **encrypted**} *auth-password* [**priv des56** {**clear** | **encrypted**} *priv-password*]]} [*access-list-name*]
5. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**snmp-server view** *view-name oid-tree* {**included** | **excluded**} **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server view view_name 1.3.6.1.2.1.1.5 included`	Creates or modifies a view record.
Step 3	**snmp-server group** *name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *view*] [**write** *view*] [**notify** *view*] [*access-list-name*] **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server group group_name v3 noauth read view_name1 write view_name2`	Configures a new SNMP group or a table that maps SNMP users to SNMP views.
Step 4	**snmp-server user** *username groupname* {**v1** | **v2c** | **v3** [**auth** {**md5** | **sha**} {**clear** | **encrypted**} *auth-password* [**priv des56** {**clear** | **encrypted**} *priv-password*]]} [*access-list-name*] **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server user noauthuser group_name v3`	Configures a new user to an SNMP group. **Note** Only one remote host can be assigned to the same username for SNMP version 3. If you configure the same username with different remote hosts, only the last username and remote host combination will be accepted and will be seen in the **show running** configuration. In the case of multiple SNMP managers, multiple unique usernames are required.
Step 5	**commit**	

Configuring SNMP Trap Notifications

This task explains how to configure the router to send SNMP trap notifications.

Note You can omit Step 3, on page 335 if you have already completed the steps documented under the Configuring SNMPv3, on page 334 task.

SUMMARY STEPS

1. **configure**
2. **snmp-server group** *name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *view*] [**write** *view*] [**notify** *view*] [*access-list-name*]
3. **snmp-server user** *username groupname* {**v1** | **v2c** | **v3** [**auth** {**md5** | **sha**} {**clear** | **encrypted**} *auth-password* [**priv des56** {**clear** | **encrypted**} *priv-password*]]} [*access-list-name*]
4. **snmp-server host** *address* [**traps**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port*] [*notification-type*]
5. **snmp-server traps** [*notification-type*]
6. **commit**
7. (Optional) **show snmp host**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**snmp-server group** *name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *view*] [**write** *view*] [**notify** *view*] [*access-list-name*] **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server group group_name v3 noauth read view_name1 write view_name2`	Configures a new SNMP group or a table that maps SNMP users to SNMP views.
Step 3	**snmp-server user** *username groupname* {**v1** | **v2c** | **v3** [**auth** {**md5** | **sha**} {**clear** | **encrypted**} *auth-password* [**priv des56** {**clear** | **encrypted**} *priv-password*]]} [*access-list-name*] **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server user noauthuser group_name v3`	Configures a new user to an SNMP group. **Note** Only one remote host can be assigned to the same username for SNMP version 3. If you configure the same username with different remote hosts, only the last username and remote host combination will be accepted and will be seen in the **show running** configuration. In the case of multiple SNMP managers, multiple unique usernames are required.
Step 4	**snmp-server host** *address* [**traps**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port*] [*notification-type*] **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server host 12.26.25.61 traps version 3 noauth userV3noauth`	Specifies SNMP trap notifications, the version of SNMP to use, the security level of the notifications, and the recipient (host) of the notifications.
Step 5	**snmp-server traps** [*notification-type*] **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server traps bgp`	Enables the sending of trap notifications and specifies the type of trap notifications to be sent. • If a trap is not specified with the *notification-type* argument, all supported trap notifications are enabled on the router. To display which trap notifications are

	Command or Action	Purpose
		available on your router, enter the **snmp-server traps ?** command.
Step 6	**commit**	
Step 7	(Optional) **show snmp host** **Example:** `RP/0/RP0/CPU0:router# show snmp host`	Displays information about the configured SNMP notification recipient (host), port number, and security model.

Setting the Contact, Location, and Serial Number of the SNMP Agent

This task explains how to set the system contact string, system location string, and system serial number of the SNMP agent.

Note The sequence in which you issue the **snmp-server** commands for this task does not matter.

SUMMARY STEPS

1. **configure**
2. (Optional) **snmp-server contact** *system-contact-string*
3. (Optional) **snmp-server location** *system-location*
4. (Optional) **snmp-server chassis-id** *serial-number*
5. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

Step 2	(Optional) **snmp-server contact** *system-contact-string*
	Example:
	`RP/0/RP0/CPU0:router(config)# snmp-server contact`
	`Dial System Operator at beeper # 27345`
		Sets the system contact string.

Step 3	(Optional) **snmp-server location** *system-location*
	Example:
	`RP/0/RP0/CPU0:router(config)# snmp-server location`
	`Building 3/Room 214`
		Sets the system location string.

Step 4	(Optional) **snmp-server chassis-id** *serial-number* **Example:**	Sets the system serial number.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config)# snmp-server chassis-id 1234456`	
Step 5	**commit**	

Defining the Maximum SNMP Agent Packet Size

This task shows how to configure the largest SNMP packet size permitted when the SNMP server is receiving a request or generating a reply.

Note The sequence in which you issue the **snmp-server** commands for this task does not matter.

SUMMARY STEPS

1. **configure**
2. (Optional) **snmp-server packetsize** *byte-count*
3. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	(Optional) **snmp-server packetsize** *byte-count* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server packetsize 1024`	Sets the maximum packet size.
Step 3	**commit**	

Changing Notification Operation Values

After SNMP notifications have been enabled, you can specify a value other than the default for the source interface, message queue length, or retransmission interval.

This task explains how to specify a source interface for trap notifications, the message queue length for each host, and the retransmission interval.

Note The sequence in which you issue the **snmp-server** commands for this task does not matter.

SUMMARY STEPS

1. **configure**
2. (Optional) **snmp-server trap-source** *type interface-path-id*
3. (Optional) **snmp-server queue-length** *length*
4. (Optional) **snmp-server trap-timeout** *seconds*
5. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	(Optional) **snmp-server trap-source** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server trap-source POS 0/0/1/0`	Specifies a source interface for trap notifications.
Step 3	(Optional) **snmp-server queue-length** *length* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server queue-length 20`	Establishes the message queue length for each notification.
Step 4	(Optional) **snmp-server trap-timeout** *seconds* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server trap-timeout 20`	Defines how often to resend notifications on the retransmission queue.
Step 5	**commit**	

Setting IP Precedence and DSCP Values

This task describes how to configure IP Precedence or IP DSCP for SNMP traffic.

Before you begin

SNMP must be configured.

SUMMARY STEPS

1. **configure**
2. Use one of the following commands:
   - **snmp-server ipv4 precedence** *value*
   - **snmp-server ipv4 dscp** *value*
3. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	Use one of the following commands: • **snmp-server ipv4 precedence** *value* • **snmp-server ipv4 dscp** *value* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server dscp 24`	Configures an IP precedence or IP DSCP value for SNMP traffic.
Step 3	**commit**	

Displaying SNMP Context Mapping

The SNMP agent serves queries based on SNMP contexts created by the client features. There is a context mapping table. Each entry in the context mapping table includes a context name, the name of the feature that created the context, and the name of the specific instance of the feature.

SUMMARY STEPS

1. **show snmp context-mapping**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show snmp context-mapping** **Example:** `RP/0/RP0/CPU0:router# show snmp context-mapping`	Displays the SNMP context mapping table.

Monitoring Packet Loss

It is possible to monitor packet loss by configuring the generation of SNMP traps when packet loss exceeds a specified threshold. The configuration described in this task enables the creation of entries in the MIB tables of the EVENT-MIB. This can then be monitored for packet loss using SNMP GET operations.

Before you begin

Note Entries created in the EVENT-MIB MIB tables using the configuration described in this task cannot be altered using an SNMP SET.

Entries to the EVENT-MIB MIB tables created using an SNMP SET cannot be altered using the configuration described in this task.

SUMMARY STEPS

1. **snmp-server mibs eventmib packet-loss** *type interface-path-id* **falling** *lower-threshold* **interval** *sampling-interval* **rising** *upper-threshold*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**snmp-server mibs eventmib packet-loss** *type interface-path-id* **falling** *lower-threshold* **interval** *sampling-interval* **rising** *upper-threshold* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server mibs eventmib packet-loss falling 1 interval 5 rising 2`	Generates SNMP EVENT-MIB traps for the interface when the packet loss exceeds the specified thresholds. Up to 100 interfaces can be monitored. **falling** *lower-threshold* —Specifies the lower threshold. When packet loss between two intervals falls below this threshold and an mteTriggerRising trap was generated previously, a SNMP mteTriggerFalling trap is generated. This trap is not generated until the packet loss exceeds the upper threshold and then falls back below the lower threshold. **interval** *sampling-interval* —Specifies how often packet loss statistics are polled. This is a value between 5 and 1440 minutes, in multiples of 5. **rising** *upper-threshold* —Specifies the upper threshold. When packet loss between two intervals increases above this threshold, an SNMP mteTriggreRising trap is generated. This trap is not generated until the packet loss drops below the lower threshold and then rises above the upper threshold.

Configuring MIB Data to be Persistent

Many SNMP MIB definitions define arbitrary 32-bit indices for their object tables. MIB implementations often do a mapping from the MIB indices to some internal data structure that is keyed by some other set of data. In these MIB tables the data contained in the table are often other identifiers of the element being modelled. For example, in the ENTITY-MIB, entries in the entPhysicalTable are indexed by the 31-bit value, entPhysicalIndex, but the entities could also be identified by the entPhysicalName or a combination of the other objects in the table.

Because of the size of some MIB tables, significant processing is required to discover all the mappings from the 32-bit MIB indices to the other data which the network management station identifies the entry. For this reason, it may be necessary for some MIB indices to be persistent across process restarts, switchovers, or device reloads. The ENTITY-MIB entPhysicalTable and CISCO-CLASS-BASED-QOS-MIB are two such MIBs that often require index values to be persistent.

Also, because of query response times and CPU utilization during CISCO-CLASS-BASED-QOS-MIB statistics queries, it is desirable to cache service policy statistics.

SUMMARY STEPS

1. (Optional) **snmp-server entityindex persist**
2. (Optional) **snmp-server mibs cbqosmib persist**

3. (Optional) **snmp-server cbqosmib cache refresh time** *time*
4. (Optional) **snmp-server cbqosmib cache service-policy count** *count*
5. **snmp-server ifindex persist**

DETAILED STEPS

	Command or Action	Purpose
Step 1	(Optional) **snmp-server entityindex persist** **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server entityindex persist`	Enables the persistent storage of ENTITY-MIB data.
Step 2	(Optional) **snmp-server mibs cbqosmib persist** **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib persist`	Enables persistent storage of the CISCO-CLASS-BASED-QOS-MIB data.
Step 3	(Optional) **snmp-server cbqosmib cache refresh time** *time* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache refresh time 45`	Enables QoS MIB caching with a specified cache refresh time.
Step 4	(Optional) **snmp-server cbqosmib cache service-policy count** *count* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server mibs cbqosmib cache service-policy count 50`	Enables QoS MIB caching with a limited number of service policies to cache.
Step 5	**snmp-server ifindex persist** **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server ifindex persist`	Enables ifIndex persistence globally on all Simple Network Management Protocol (SNMP) interfaces.

Configuring LinkUp and LinkDown Traps for a Subset of Interfaces

By specifying a regular expression to represent the interfaces for which you are interested in setting traps, you can enable or disable linkUp and linkDown traps for a large number of interfaces simultaneously.

Before you begin

SNMP must be configured.

SUMMARY STEPS

1. **configure**
2. **snmp-server interface subset** *subset-number* **regular-expression** *expression*
3. **notification linkupdown disable**
4. **commit**
5. (Optional) **show snmp interface notification subset** *subset-number*
6. (Optional) **show snmp interface notification regular-expression** *expression*
7. (Optional) **show snmp interface notification** *type interface-path-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**snmp-server interface subset** *subset-number* **regular-expression** *expression* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server interface subset 10 regular-expression "^Gig[a-zA-Z]+[0-9/]+\." RP/0/RP0/CPU0:router(config-snmp-if-subset)#`	Enters snmp-server interface mode for the interfaces identified by the regular expression. The subset-number argument identifies the set of interfaces, and also assigns a priority to the subset in the event that an interface is included in more than one subset. Lower numbers have higher priority and their configuration takes precedent over interface subsets with higher numbers. The *expression* argument must be entered surrounded by double quotes. Refer to the *Understanding Regular Expressions, Special Characters, and Patterns* module in *Cisco IOS XR Getting Started Guide for the Cisco CRS Router* for more information regarding regular expressions.
Step 3	**notification linkupdown disable** **Example:** `RP/0/RP0/CPU0:router(config-snmp-if-subset)# notification linkupdown disable`	Disables linkUp and linkDown traps for all interfaces being configured. To enable previously disabled interfaces, use the **no** form of this command.
Step 4	**commit**	
Step 5	(Optional) **show snmp interface notification subset** *subset-number* **Example:** `RP/0/RP0/CPU0:router# show snmp interface notification subset 10`	Displays the linkUp and linkDown notification status for all interfaces identified by the subset priority.
Step 6	(Optional) **show snmp interface notification regular-expression** *expression* **Example:** `RP/0/RP0/CPU0:router# show snmp interface`	Displays the linkUp and linkDown notification status for all interfaces identified by the regular expression.

	Command or Action	Purpose
	`notification regular-expression "^Gig[a-zA-Z]+[0-9/]+\."`	
Step 7	(Optional) **show snmp interface notification** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router# show snmp interface notification tengige 0/4/0/3.10`	Displays the linkUp and linkDown notification status for the specified interface.

Configuration Examples for Implementing SNMP

Configuring SNMPv3: Examples

Setting an Engine ID

This example shows how to set the identification of the local SNMP engine:

```
snmp-server engineID local 00:00:00:09:00:00:00:a1:61:6c:20:61
```

Note After the engine ID has been configured, the SNMP agent restarts.

Verifying the Identification of the Local SNMP Engines

This example shows how to verify the identification of the local SNMP engine:

```
config
  show snmp engineid

  SNMP engineID 00000009000000a1ffffffff
```

Creating a View

There are two ways to create a view:

- You can include the object identifier (OID) of an ASN.1 subtree of a MIB family from a view by using the **included** keyword of the **snmp-server view** command.
- You can exclude the OID subtree of the ASN.1 subtree of a MIB family from a view by using the **excluded** keyword of the **snmp-server view** command.

This example shows how to create a view that includes the sysName (1.3.6.1.2.1.1.5) object:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1.5 included
```

This example shows how to create a view that includes all the OIDs of a system group:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
```

This example shows how to create a view that includes all the OIDs under the system group except the sysName object (1.3.6.1.2.1.1.5), which has been excluded:

```
config
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1.5 excluded
```

Verifying Configured Views

This example shows how to display information about the configured views:

```
RP/0/RP0/CPU0:router# show snmp view

  v1default 1.3.6.1 - included nonVolatile active
  SNMP_VIEW1 1.3.6.1.2.1.1 - included nonVolatile active
  SNMP_VIEW1 1.3.6.1.2.1.1.5 - excluded nonVolatile active
```

Creating Groups

If you do not explicitly specify a notify, read, or write view, the Cisco IOS XR software uses the v1 default (1.3.6.1). This example shows how to create a group that utilizes the default view:

```
RP/0/RP0/CPU0:router(config)# snmp-server group group-name v3 auth
```

The following configuration example shows how to create a group that has read access to all the OIDs in the system except the sysUpTime object (1.3.6.1.2.1.1.3), which has been excluded from the view applied to the group, but write access only to the sysName object (1.3.6.1.2.1.1.5):

```
!
  snmp-server view view_name1 1.3.6.1.2.1.1 included
  snmp-server view view_name1 1.3.6.1.2.1.1.3 excluded
  snmp-server view view_name2 1.3.6.1.2.1.1.5 included
  snmp-server group group_name1 v3 auth read view_name1 write view_name2
  !
```

Verifying Groups

This example shows how to verify the attributes of configured groups:

```
RP/0/RP0/CPU0:router# show snmp group

  groupname: group_name1                  security model:usm
  readview : view_name1                   writeview: view_name2
  notifyview: v1default
  row status: nonVolatile
```

Creating and Verifying Users

Given the following SNMPv3 view and SNMPv3 group configuration:

```
!
  snmp-server view view_name 1.3.6.1.2.1.1 included
  snmp-server group group_name v3 noauth read view_name write view-name
  !
```

This example shows how to create a noAuthNoPriv user with read and write view access to a system group:

```
config
  snmp-server user noauthuser group_name v3
```

Note The user must belong to a noauth group before a noAuthNoPriv user can be created.

Only one remote host can be assigned to the same username for SNMP version 3. If you configure the same username with different remote hosts, only the last username and remote host combination will be accepted and will be seen in the show running configuration. In the case of multiple SNMP managers, multiple unique usernames are required.

This example shows the same username case which only the last configuration will be accepted:

```
snmp-server user username  nervectrgrp remote 10.69.236.146 udp-port 162 v3 auth sha
<password> priv aes 128 <password>
snmp-server user username  nervectrgrp remote 10.214.127.2 udp-port 162 v3 auth sha <password>
 priv aes 128  <password>
snmp-server user username  nervectrgrp remote 10.69.236.147 udp-port 162 v3 auth sha
<password> priv aes 128 <password>
RP/0/RP0/CPU0:router# show run snmp-server user

  snmp-server user username nervectrgrp remote 10.69.236.147 udp-port 162 v3 auth sha
encrypted <password> priv aes 128 encrypted <password>
```

This example shows all 3 hosts for username1, username2, and username3 will be accepted.

:

```
snmp-server user username1  nervectrgrp remote 10.69.236.146 udp-port 162 v3 auth sha
<password> priv aes 128 <password>
snmp-server user username2  nervectrgrp remote 10.214.127.2 udp-port 162 v3 auth sha
<password> priv aes 128  <password>
snmp-server user username3  nervectrgrp remote 10.69.236.147 udp-port 162 v3 auth sha
<password> priv aes 128 <password>
```

```
RP/0/RP0/CPU0:router# show run snmp-server user

  snmp-server user batmanusr1 nervectrgrp remote 10.69.236.146 udp-port 162 v3 auth sha
encrypted <password> priv aes 128 encrypted <password>
  snmp-server user batmanusr2 nervectrgrp remote 10.214.127.2 udp-port 162 v3 auth sha
encrypted <password> priv aes 128 encrypted <password>
  snmp-server user batmanusr3 nervectrgrp remote 10.69.236.147 udp-port 162 v3 auth sha
encrypted <password> priv aes 128 encrypted <password>
```

This example shows how to verify the attributes that apply to the SNMP user:

```
RP/0/RP0/CPU0:router# show snmp user

  User name: noauthuser
  Engine ID: localSnmpID
  storage-type: nonvolatile active
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```
!
  snmp-server view SNMP_VIEW1 1.3.6.1.2.1.1 included
  snmp-server group SNMP_GROUP1 v3 auth notify SNMP_VIEW1 read SNMP_VIEW1 write SNMP_VIEW1
 !
```

This example shows how to create a user with authentication (including encryption), read, and write view access to a system group:

```
config
  snmp-server user userv3authpriv SNMP_GROUP1 v3 auth md5 password123 priv aes 128 password123
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```
!
  snmp-server view view_name 1.3.6.1.2.1.1 included
  snmp group group_name v3 priv read view_name write view_name
  !
```

This example shows how to create authNoPriv user with read and write view access to a system group:

```
RP/0/RP0/CPU0:router(config)# snmp-server user authuser group_name v3 auth md5 clear
auth_passwd
```

Note Because the group is configured at a security level of Auth, the user must be configured as “auth” at a minimum to access this group (“priv” users could also access this group). The authNoPriv user configured in this group, authuser, must supply an authentication password to access the view. In the example, auth_passwd is set as the authentication password string. Note that **clear** keyword is specified before the auth_passwd password string. The **clear** keyword indicates that the password string being supplied is unencrypted.

This example shows how to verify the attributes that apply to SNMP user:

```
RP/0/RP0/CPU0:router# show snmp user

  User name: authuser
  Engine ID: localSnmpID
  storage-type: nonvolatile active
```

Given the following SNMPv3 view and SNMPv3 group configuration:

```
!
  snmp view view_name 1.3.6.1.2.1.1 included
  snmp group group_name v3 priv read view_name write view_name
  !
```

This example shows how to create an authPriv user with read and write view access to a system group:

```
config
  snmp-server user privuser group_name v3 auth md5 clear auth_passwd priv des56 clear
priv_passwd
```

Note Because the group has a security level of Priv, the user must be configured as a "priv" user to access this group. In this example, the user, privuser, must supply both an authentication password and privacy password to access the OIDs in the view.

This example shows how to verify the attributes that apply to the SNMP user:

```
RP/0/RP0/CPU0:router# show snmp user

  User name: privuser
  Engine ID: localSnmpID
  storage-type: nonvolatile active
```

Configuring Trap Notifications: Example

The following example configures an SNMP agent to send out different types of traps. The configuration includes a v2c user, a noAuthNoPriv user, anauthNoPriv user, and an AuthPriv user.

Note The default User Datagram Protocol (UDP) port is 161. If you do not a specify a UDP port with the **udp-port** keyword and *port* argument, then the configured SNMP trap notifications are sent to port 161.

```
!
  snmp-server host 10.50.32.170 version 2c public udp-port 2345
  snmp-server host 10.50.32.170 version 3 auth userV3auth udp-port 2345
```

```
 snmp-server host 10.50.32.170 version 3 priv userV3priv udp-port 2345
 snmp-server host 10.50.32.170 version 3 noauth userV3noauth udp-port 2345
 snmp-server user userv2c groupv2c v2c
 snmp-server user userV3auth groupV3auth v3 auth md5 encrypted 140F0A13
 snmp-server user userV3priv groupV3priv v3 auth md5 encrypted 021E1C43 priv des56 encrypted
1110001C
 snmp-server user userV3noauth groupV3noauth v3 LROwner
 snmp-server view view_name 1.3 included
 snmp-server community public RW
 snmp-server group groupv2c v2c read view_name
 snmp-server group groupV3auth v3 auth read view_name
 snmp-server group groupV3priv v3 priv read view_name
 snmp-server group groupV3noauth v3 noauth read view_name
 !
```

This example shows how to verify the configuration SNMP trap notification recipients host, the recipients of SNMP trap notifications. The output displays the following information:

- IP address of the configured notification host
- UDP port where SNMP notification messages are sent
- Type of trap configured
- Security level of the configured user
- Security model configured

```
config
  show snmp host

  Notification host: 10.50.32.170 udp-port: 2345 type: trap
  user: userV3auth security model: v3 auth

  Notification host: 10.50.32.170 udp-port: 2345 type: trap
  user: userV3noauth security model: v3 noauth

  Notification host: 10.50.32.170 udp-port: 2345 type: trap
  user: userV3priv security model: v3 priv

  Notification host: 10.50.32.170 udp-port: 2345 type: trap
  user: userv2c security model: v2c
```

Setting an IP Precedence Value for SNMP Traffic: Example

The following example shows how to set the SNMP IP Precedence value to 7:

```
configure
  snmp-server ipv4 precedence 7
  exit

  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y
```

Setting an IP DSCP Value for SNMP Traffic: Example

The following example shows how to set the IP DSCP value of SNMP traffic to 45:

```
configure
  snmp-server ipv4 dscp 45
  exit

  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]: y
```

Additional References

The following sections provide references related to Implementing SNMP on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Cisco IOS XR SNMP commands	*SNMP Server Commands on the Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*
MIB information	*Cisco CRS and Cisco XR 12000 Series Router MIB Overview*
Cisco IOS XR commands	*Cisco IOS XR Commands Master List for the Cisco CRS Router*
Getting started with Cisco IOS XR software	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*
Information about user groups and task IDs	*Configuring AAA Services on the Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*
Cisco IOS XR Quality of Service	*Modular QoS Configuration Guide for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

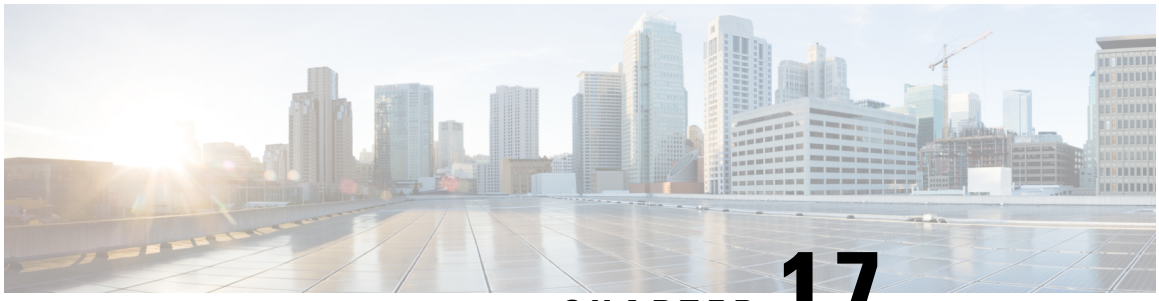
MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
RFC 3411	*An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*
RFC 3412	*Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*
RFC 3413	*Simple Network Management Protocol (SNMP) Applications*
RFC 3414	*User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3416	*Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*
RFC 3417	*Transport Mappings for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 17

Configuring Periodic MIB Data Collection and Transfer

This document describes how to periodically transfer selected MIB data from your router to a specified Network Management System (NMS). The periodic MIB data collection and transfer feature is also known as bulk statistics.

Table 39: Feature History for Periodic MIB Data Collection and Transfer

Release	Modification
Release 4.2.0	The periodic MIB data collection and transfer feature was introduced and supported the IF-MIB only.
Release 4.2.1	Additional MIBs were supported.

This module contains the following topics:

Prerequisites for Periodic MIB Data Collection and Transfer

To use periodic MIB data collection and transfer, you should be familiar with the Simple Network Management Protocol (SNMP) model of management information. You should also know what MIB information you want to monitor on your network devices, and the OIDs or object names for the MIB objects to be monitored.

Information About Periodic MIB Data Collection and Transfer

SNMP Objects and Instances

A type (or class) of SNMP management information is called an object. A specific instance from a type of management information is called an object instance (or SNMP variable). To configure a bulk statistics collection, you must specify the object types to be monitored using a bulk statistics object list and the specific instances of those objects to be collected using a bulk statistics schema.

MIBs, MIB tables, MIB objects, and object indices can all be specified using a series of numbers called an object identifier (OID). OIDs are used in configuring a bulk statistics collection in both the bulk statistics object lists (for general objects) and in the bulk statistics schemas (for specific object instances).

Bulk Statistics Object Lists

To group the MIB objects to be polled, you need to create one or more object lists. A bulk statistics object list is a user-specified set of MIB objects that share the same MIB index. Object lists are identified using a name that you specify. Named bulk statistics object lists allow the same configuration to be reused in different bulk statistics schemas.

All the objects in an object list must share the same MIB index. However, the objects do not need to be in the same MIB and do not need to belong to the same MIB table. For example, it is possible to group ifInOctets and a CISCO-IF-EXTENSION-MIB object in the same schema, because the containing tables for both objects are indexed by the ifIndex.

Bulk Statistics Schemas

Data selection for the Periodic MIB Data Collection and Transfer Mechanism requires the definition of a schema with the following information:

- Name of an object list.
- Instance (specific instance or series of instances defined using a wild card) that needs to be retrieved for objects in the specified object list.
- How often the specified instances need to be sampled (polling interval). The default polling interval is 5 minutes.

A bulk statistics schema is also identified using a name that you specify. This name is used when configuring the transfer options.

Bulk Statistics Transfer Options

After configuring the data to be collected, a single virtual file (VFile or *bulk statistics file*) with all collected data is created. This file can be transferred to a network management station using FTP or TFTP. You can specify how often this file should be transferred. The default transfer interval is once every 30 minutes. You can also configure a secondary destination for the file to be used if, for whatever reason, the file cannot be transferred to the primary network management station.

The value of the transfer interval is also the collection period (collection interval) for the local bulk statistics file. After the collection period ends, the bulk statistics file is frozen, and a new local bulk statistics file is created for storing data. The frozen bulk statistics file is then transferred to the specified destination.

By default, the local bulk statistics file is deleted after successful transfer to an network management station.

Benefits of Periodic MIB Data Collection and Transfer

Periodic MIB data collection and transfer (bulk statistics feature) allows many of the same functions as the bulk file MIB (CISCO-BULK-FILE-MIB.my), but offers some key advantages. The main advantage is that this feature can be configured through the CLI and does not require an external monitoring application.

Periodic MIB data collection and transfer is mainly targeted for medium to high-end platforms that have sufficient local storage (volatile or permanent) to store bulk statistics files. Locally storing bulk statistics files helps minimize loss of data during temporary network outages.

This feature also has more powerful data selection features than the bulk file MIB; it allows grouping of MIB objects from different tables into data groups (object lists). It also incorporates a more flexible instance selection mechanism, where the application is not restricted to fetching an entire MIB table.

4arg

4arg is Cisco's implementation of Object Size Checking (OSC). OSC is a useful static analysis utility and a critical runtime defense for the detection and prevention of buffer overflows. 4arg captures buffer overflows (at runtime) that are otherwise undetected by code analysis tools and human review. 4arg also carries a reporting and logging component.

4arg refers to a theoretical fourth argument to a string copy function representing the destination buffer size.

4arg Messages: Example

An example of a 4arg message:

```
RP/0/RP0/CPU0:router:Dec 1 12:00:00.802 : foo[123]:
%OS-DATACORRUPTION-1-DATAINCONSISTENCY : copy error  : pkg/bin/foo :
(PID=12345) :  -Traceback= 4bd43404 4bac7e04 4000c100
```

Note If you witness a traceback, please report the incident to Cisco TAC as soon as possible. Include the log message exactly as printed in the console.

How to Configure Periodic MIB Data Collection and Transfer

Configuring a Bulk Statistics Object List

The first step in configuring the Periodic MIB Data Collection and Transfer Mechanism is to configure one or more object lists.

SUMMARY STEPS

1. **configure**
2. **snmp-server mib bulkstat object-list** *list-name*
3. **add** {**oid** | *object-name*}
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**snmp-server mib bulkstat object-list** *list-name* **Example:** `snmp-server mib bulkstat object-list ifMib`	Defines an SNMP bulk statistics object list and enters bulk statistics object list configuration mode.

Step 3	**add** {**oid**	*object-name*}
		Example:
		`RP/0/RP0/CPU0:router(config-bulk-objects)# add 1.3.6.1.2.1.2.2.1.11`
		`RP/0/RP0/CPU0:router(config-bulk-objects)# add ifAdminStatus`
		`RP/0/RP0/CPU0:router(config-bulk-objects)# add ifDescr`
			Adds a MIB object to the bulk statistics object list. Repeat as desired until all objects to be monitored in this list are added.
			Note All the objects in a bulk statistics object list have to be indexed by the same MIB index. However, the objects in the object list do not need to belong to the same MIB or MIB table.
			When specifying an object name instead of an OID (using the add command), only object names with mappings shown in the **show snmp mib object** command output can be used.

Step 4	**commit**	

What to do next

Configure a bulk statistics schema.

Configuring a Bulk Statistics Schema

The second step in configuring periodic MIB data collection and transfer is to configure one or more schemas.

Before you begin

The bulk statistics object list to be used in the schema must be defined.

SUMMARY STEPS

1. **configure**
2. **snmp-server mib bulkstat schema** *schema-name*
3. **object-list** *list-name*
4. Do one of the following:

- **instance exact** {**interface** *interface-id* [**sub-if**] | **oid** *oid*}
- **instance wild** {**interface** *interface-id* [**sub-if**] | **oid** *oid*}
- **instance range start** *oid* **end** *oid*
- **instance repetition** *oid* **max** *repeat-number*

5. **poll-interval** *minutes*
6. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	

Step 2	**snmp-server mib bulkstat schema** *schema-name*
	Example:
	`RP/0/RP0/CPU0:router(config)# snmp-server mib bulkstat schema intE0`
	`RP/0/RP0/CPU0:router(config-bulk-sc)#`
		Names the bulk statistics schema and enters bulk statistics schema mode.

Step 3	**object-list** *list-name* **Example:** `RP/0/RP0/CPU0:router(config-bulk-sc)# object-list ifMib`	Specifies the bulk statistics object list to be included in this schema. Specify only one object list per schema. If multiple object-list commands are executed, the earlier ones are overwritten by newer commands.

Step 4	Do one of the following:
	• **instance exact** {**interface** *interface-id* [**sub-if**] | **oid** *oid*}
	• **instance wild** {**interface** *interface-id* [**sub-if**] | **oid** *oid*}
	• **instance range start** *oid* **end** *oid*
	• **instance repetition** *oid* **max** *repeat-number*
	Example:
	`RP/0/RP0/CPU0:router(config-bulk-sc)# instance wild oid 1`
	or
	`RP/0/RP0/CPU0:router(config-bulk-sc)# instance exact interface FastEthernet 0/1.25`
	or
	`RP/0/RP0/CPU0:router(config-bulk-sc)# instance range start 1 end 2`
	or
	`RP/0/RP0/CPU0:router(config-bulk-sc)# instance repetition 1 max 4`
		Specifies the instance information for objects in this schema:
		• The **instance exact** command indicates that the specified instance, when appended to the object list, represents the complete OID.
		• The **instance wild** command indicates that all subindices of the specified OID belong to this schema. The wild keyword allows you to specify a partial, "wild carded" instance.
		• The **instance range** command indicates a range of instances on which to collect data.
		• The **instance repetition** command indicates data collection to repeat for a certain number of instances of a MIB object.
		Note Only one **instance** command can be configured per schema. If multiple **instance** commands are executed, the earlier ones are overwritten by new commands.

	Command or Action	Purpose
Step 5	**poll-interval** *minutes* **Example:** `RP/0/RP0/CPU0:router(config-bulk-sc)# poll-interval 10`	Sets how often data should be collected from the object instances specified in this schema, in minutes. The default is once every 5 minutes. The valid range is from 1 to 20000.
Step 6	**commit**	

What to do next

Configure the bulk statistics transfer options.

Configuring Bulk Statistics Transfer Options

The final step in configuring periodic MIB data collection and transfer is to configure the transfer options. The collected MIB data are kept in a local file-like entity called a VFile (virtual file, referred to as a bulk statistics file in this document). This file can be transferred to a remote network management station at intervals you specify.

Before you begin

The bulk statistics object lists and bulk statistics schemas must be defined before configuring the bulk statistics transfer options.

SUMMARY STEPS

1. **configure**
2. **snmp-server mib bulkstat transfer-id** *transfer-id*
3. **buffer-size** *bytes*
4. **format** {**bulkBinary** | **bulkASCII** | **schemaASCII**}
5. **schema** *schema-name*
6. **transfer-interval** *minutes*
7. **url primary** *url*
8. **url secondary** *url*
9. **retry** *number*
10. **retain** *minutes*
11. **enable**
12. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**snmp-server mib bulkstat transfer-id** *transfer-id* **Example:** `RP/0/RP0/CPU0:router(config)# snmp-server mib bulkstat transfer bulkstat1`	Identifies the transfer configuration with a name (*transfer-id* argument) and enters bulk statistics transfer configuration mode.

	Command or Action	Purpose
Step 3	**buffer-size** *bytes* **Example:** `RP/0/RP0/CPU0:router(config-bulk-tr)# buffersize 3072`	(Optional) Specifies the maximum size for the bulk statistics data file, in bytes. The valid range is from 1024 to 2147483647 bytes. The default buffer size is 2048 bytes. **Note** If the maximum buffer size for a bulk statistics file is reached before the transfer interval time expires, all additional data received is deleted. To correct this behavior, you can decrease the polling frequency, or increase the size of the bulk statistics buffer.
Step 4	**format** {**bulkBinary** | **bulkASCII** | **schemaASCII**} **Example:** `RP/0/RP0/CPU0:router(config-bulk-tr)# format schemaASCII`	(Optional) Specifies the format of the bulk statistics data file (VFile). The default is schemaASCII. **Note** Transfers can only be performed using schemaASCII (cdcSchemaASCII) format. SchemaASCII is a human-readable format that contains parser-friendly hints for parsing data values.

Step 5	**schema** *schema-name*
	Example:
	`RP/0/RP0/CPU0:router(config-bulk-tr)# schema ATM2/0-IFMIB`
	`RP/0/RP0/CPU0:router(config-bulk-tr)# schema ATM2/0-CAR`
	`RP/0/RP0/CPU0:router(config-bulk-tr)# schema Ethernet2/1-IFMIB`
		Specifies the bulk statistics schema to be transferred. Repeat this command as desired. Multiple schemas can be associated with a single transfer configuration; all collected data are placed in a single bulk data file (VFile).

Step 6	**transfer-interval** *minutes* **Example:** RP/0/RP0/CPU0:router `RP/0/RP0/CPU0:router(config-bulk-tr)# transfer-interval 20`	(Optional) Specifies how often the bulk statistics file are transferred, in minutes. The default value is once every 30 minutes. The transfer interval is the same as the collection interval.
Step 7	**url primary** *url* **Example:** `RP/0/RP0/CPU0:router(config-bulk-tr)# url primary ftp://user:password@host/folder/bulkstat1`	Specifies the network management system (host) that the bulk statistics data file is transferred to, and the protocol to use for transfer. The destination is specified as a Uniform Resource Locator (URL). FTP or TFTP can be used for the bulk statistics file transfer.
Step 8	**url secondary** *url* **Example:** `RP/0/RP0/CPU0:router(config-bulk-tr)# url secondary tftp://10.1.0.1/tftpboot/user/bulkstat1`	(Optional) Specifies a backup transfer destination and protocol for use in the event that transfer to the primary location fails. FTP or TFTP can be used for the bulk statistics file transfer.

	Command or Action	Purpose
Step 9	**retry** *number* **Example:** `RP/0/RP0/CPU0:router(config-bulk-tr)# retry 1`	(Optional) Specifies the number of transmission retries. The default value is 0 (in other words, no retries). If an attempt to send the bulk statistics file fails, the system can be configured to attempt to send the file again using this command. One retry includes an attempt first to the primary destination then, if the transmission fails, to the secondary location. For example, if the retry value is 1, an attempt is made first to the primary URL, then to the secondary URL, then to the primary URL again, then to the secondary URL again. The valid range is from 0 to 100. If all retries fail, the next normal transfer occurs after the configured transfer-interval time.
Step 10	**retain** *minutes* **Example:** `RP/0/RP0/CPU0:router(config-bulk-tr)# retain 60`	(Optional) Specifies how long the bulk statistics file should be kept in system memory, in minutes, after the completion of the collection interval and a transmission attempt is made. The default value is 0. Zero (0) indicates that the file is deleted immediately after the transfer is attempted. The valid range is from 0 to 20000. **Note** If the retry command is used, you should configure a retain interval larger than 0. The interval between retries is the retain interval divided by the retry number. For example, if **retain 10** and **retry 2** are configured, two retries are attempted once every 5 minutes. Therefore, if retain 0 is configured, no retries are attempted.
Step 11	**enable** **Example:** `RP/0/RP0/CPU0:router(config-bulk-tr)# enable`	Begins the bulk statistics data collection and transfer process for this configuration. • For successful execution of this action, at least one schema with non-zero number of objects must be configured. • Periodic collection and file transfer begins only if this command is configured. Conversely, the **no enable** command stops the collection process. A subsequent **enable** starts the operations again. • Each time the collection process is started using the **enable** command, data is collected into a new bulk statistics file. When the **no enable** command is used, the transfer process for any collected data immediately begins (in other words, the existing bulk statistics file is transferred to the specified management station).
Step 12	**commit**	

What to do next

Note If the maximum buffer size for a bulk statistics file is reached before the transfer interval time expires, the transfer operation is still initiated, but any bulk statistics data received after the file was full, and before it was transferred, are deleted. To correct this behavior, you can decrease the polling frequency, or increase the size of the bulk statistics buffer.

If **retain 0** is configured, no retries are attempted. This is because the interval between retries is the retain value divided by the retry value. For example, if **retain 10** and **retry 2** are configured, retries are attempted once every 5 minutes. Therefore, if you configure the retry command, you should also configure an appropriate value for the retain command.

Monitoring Periodic MIB Data Collection and Transfer

SUMMARY STEPS

1. **show snmp mib bulkstat transfer** *transfer-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show snmp mib bulkstat transfer** *transfer-name*	(Optional) The show command for this feature lists all bulk statistics virtual files (VFiles) on the system that have finished collecting data. (Data files that are not complete are not displayed.) The output lists all of the completed local bulk statistics files, the remaining time left before the bulk statistics file is deleted (remaining retention period), and the state of the bulk statistics file. The "STATE" of the bulk statistics file is one of the following: • Queued--Indicates that the data collection for this bulk statistics file is completed (in other words, the transfer interval has been met) and that the bulk statistics file is waiting for transfer to the configured destination(s). • Retry--Indicates that one or more transfer attempts have failed and that the file transfer will be attempted again. The number of retry attempts remaining are displayed in parenthesis. • Retained--Indicates that the bulk statistics file has either been successfully transmitted or that the configured number of retries have been completed.

	Command or Action	Purpose
		To display only the status of a named transfer (as opposed to all configured transfers), specify the name of the transfer in the transfer-name argument.

show snmp mib bulkstat transfer Sample Output

```
RP/0/RP0/CPU0:router# show snmp mib bulkstat transfer

Transfer Name : ifmib
 Retained files

 File Name                      : Time Left (in seconds)   :STATE
 ------------------------------------------------------------------------
 ifmib_Router_020421_100554683 : 173 : Retry (2 Retry attempt(s) Left)
```

Periodic MIB Data Collection and Transfer: Example

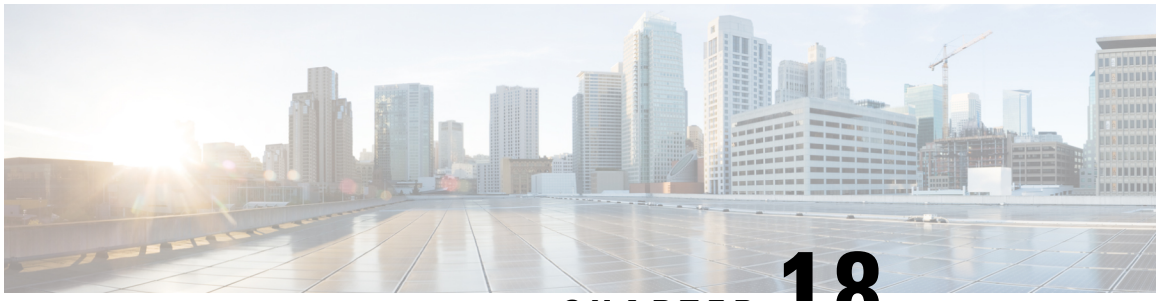
This example shows how to configure periodic MIB data collection and transfer:

```
snmp-server mib bulkstat object-list cempo
add cempMemPoolName
add cempMemPoolType
!
snmp-server mib bulkstat schema cempWild
object-list cempo
instance wild oid 8695772
poll-interval 1
!
snmp-server mib bulkstat schema cempRepeat
object-list cempo
instance repetition 8695772.1 max 4294967295
poll-interval 1
!
snmp-server mib bulkstat transfer-id cempt1
enable
url primary tftp://223.255.254.254/auto/tftp-sjc-users3/dseeniva/dumpdcm
schema cempWild
schema cempRepeat
transfer-interval 2
!
```

This example shows sample bulk statistics file content:

```
Schema-def cempt1.cempWild "%u, %s, %s, %d" Epochtime instanceoid
            1.3.6.1.4.1.9.9.221.1.1.1.1.3 1.3.6.1.4.1.9.9.221.1.1.1.1.2
cempt1.cempWild: 1339491515, 8695772.1, processor, 2
cempt1.cempWild: 1339491515, 8695772.2, reserved, 11
cempt1.cempWild: 1339491515, 8695772.3, image, 12
cempt1.cempWild: 1339491575, 8695772.1, processor, 2
cempt1.cempWild: 1339491575, 8695772.2, reserved, 11
cempt1.cempWild: 1339491575, 8695772.3, image, 12
Schema-def cempt1.cempRepeat "%u, %s, %s, %d" Epochtime instanceoid
```

```
            1.3.6.1.4.1.9.9.221.1.1.1.1.3 1.3.6.1.4.1.9.9.221.1.1.1.1.2
cempt1.cempRepeat: 1339491515, 8695772.1, processor, 2
cempt1.cempRepeat: 1339491515, 8695772.2, reserved, 11
cempt1.cempRepeat: 1339491515, 8695772.3, image, 12
cempt1.cempRepeat: 1339491515, 26932192.1, processor, 2
cempt1.cempRepeat: 1339491515, 26932192.2, reserved, 11
cempt1.cempRepeat: 1339491515, 26932192.3, image, 12
cempt1.cempRepeat: 1339491515, 35271015.1, processor, 2
cempt1.cempRepeat: 1339491515, 35271015.2, reserved, 11
cempt1.cempRepeat: 1339491515, 35271015.3, image, 12
cempt1.cempRepeat: 1339491515, 36631989.1, processor, 2
cempt1.cempRepeat: 1339491515, 36631989.2, reserved, 11
cempt1.cempRepeat: 1339491515, 36631989.3, image, 12
cempt1.cempRepeat: 1339491515, 52690955.1, processor, 2
cempt1.cempRepeat: 1339491515, 52690955.2, reserved, 11
cempt1.cempRepeat: 1339491515, 52690955.3, image, 12
```



CHAPTER 18

Configuring Cisco Discovery Protocol

Cisco Discovery Protocol (CDP) is a media- and protocol-independent protocol that runs on all Cisco-manufactured equipment including routers, bridges, access and communication servers, and switches. Using CDP, you can view information about all the Cisco devices that are directly attached to the device.

This module describes the new and revised tasks you need to implement CDP on your Cisco IOS XR network.

For more information about CDP on the Cisco IOS XR software and complete descriptions of the CDP commands listed in this module, refer to Related Documents, on page 372. To locate documentation for other commands that might appear in the course of running a configuration task, search online in *Cisco IOS XR Commands Master List for the Cisco CRS Router*.

Table 40: Feature History for Implementing CDP on Cisco IOS XR Software

Release	Modification
Release 2.0	This feature was introduced.

This module contains the following topics:

Prerequisites for Implementing CDP

You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Information About Implementing CDP

CDP is primarily used to obtain protocol addresses of neighboring devices and discover the platform of those devices. CDP can also be used to display information about the interfaces your router uses. CDP is media-

and protocol-independent, and runs on all equipment manufactured by Cisco, including routers, bridges, access servers, and switches.

Use of SNMP with the CDP MIB allows network management applications to learn the device type and the SNMP agent address of neighboring devices and to send SNMP queries to those devices. CDP uses the CISCO-CDP-MIB.

CDP runs on all media that support Subnetwork Access Protocol (SNAP), including LAN, Frame Relay, and ATM physical media. CDP runs over the data link layer only. Therefore, two systems that support different network-layer protocols can learn about each other.

Each device configured for CDP sends periodic messages, known as *advertisements*, to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or hold-time, information, which indicates the length of time a receiving device holds CDP information before discarding it. Each device also listens to the periodic CDP messages sent by others to learn about neighboring devices and determine when their interfaces to the media go up or down.

CDP Version-2 (CDPv2) is the most recent release of the protocol and provides more intelligent device tracking features. These features include a reporting mechanism that allows for more rapid error tracking, thereby reducing costly downtime. Reported error messages can be sent to the console or to a logging server, and can cover instances of unmatching native VLAN IDs (IEEE 802.1Q) on connecting ports, and unmatching port duplex states between connecting devices.

CDPv2 **show** commands can provide detailed output on VLAN Trunking Protocol (VTP) management domain and duplex modes of neighbor devices, CDP-related counters, and VLAN IDs of connecting ports.

Type-length-value fields (TLVs) are blocks of information embedded in CDP advertisements. Table 41: Type-Length-Value Definitions for CDPv2, on page 366 summarizes the TLV definitions for CDP advertisements.

Table 41: Type-Length-Value Definitions for CDPv2

TLV	Definition
Device-ID TLV	Identifies the device name in the form of a character string.
Address TLV	Contains a list of network addresses of both receiving and sending devices.
Port-ID TLV	Identifies the port on which the CDP packet is sent.
Capabilities TLV	Describes the functional capability for the device in the form of a device type; for example, a switch.
Version TLV	Contains information about the software release version on which the device is running.
Platform TLV	Describes the hardware platform name of the device, for example, Cisco 4500.
VTP Management Domain TLV	Advertises the system's configured VTP management domain name-string. Used by network operators to verify VTP domain configuration in adjacent network nodes.

TLV	Definition
Native VLAN TLV	Indicates, per interface, the assumed VLAN for untagged packets on the interface. CDP learns the native VLAN for an interface. This feature is implemented only for interfaces that support the IEEE 802.1Q protocol.
Full/Half Duplex TLV	Indicates status (duplex configuration) of CDP broadcast interface. Used by network operators to diagnose connectivity problems between adjacent network elements.

How to Implement CDP on Cisco IOS XR Software

Enabling CDP

To enable CDP, you must first enable CDP globally on the router and then enable CDP on a per-interface basis. This task explains how to enable CDP globally on the router and then enable CDP on an interface.

SUMMARY STEPS

1. **configure**
2. **cdp**
3. **interface** *type interface-path-id*
4. **cdp**
5. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**cdp** **Example:** `RP/0/RP0/CPU0:router(config)# cdp`	Enables CDP globally.
Step 3	**interface** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router(config)# interface pos 0/0/0/1`	Enters interface configuration mode.
Step 4	**cdp** **Example:** `RP/0/RP0/CPU0:router(config-if)# cdp`	Enables CDP on an interface.
Step 5	**commit**	

Modifying CDP Default Settings

This task explains how to modify the default version, hold-time setting, and timer settings.

Note The commands can be entered in any order.

SUMMARY STEPS

1. **configure**
2. **cdp advertise v1**
3. **cdp holdtime** *seconds*
4. **cdp timer** *seconds*
5. **commit**
6. (Optional) **show cdp**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**cdp advertise v1** **Example:** `RP/0/RP0/CPU0:router(config)# cdp advertise v1`	Configures CDP to use only version 1 (CDPv1) in communicating with neighboring devices. • By default, when CDP is enabled, the router sends CDPv2 packets. CDP also sends and receives CDPv1 packets if the device with which CDP is interacting does not process CDPv2 packets. • In this example, the router is configured to send and receive only CDPv1 packets.
Step 3	**cdp holdtime** *seconds* **Example:** `RP/0/RP0/CPU0:router(config)# cdp holdtime 30`	Specifies the amount of time that the receiving networking device will hold a CDP packet sent from the router before discarding it. • By default, when CDP is enabled, the receiving networking device holds a CDP packet for 180 seconds before discarding it. **Note** The CDP hold time must be set to a higher number of seconds than the time between CDP transmissions, which is set with the **cdp timer** command. • In this example, the value of hold-time for the *seconds* argument is set to 30.
Step 4	**cdp timer** *seconds* **Example:**	Specifies the frequency at which CDP update packets are sent.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config)# cdp timer 20`	• By default, when CDP is enabled, CDP update packets are sent at a frequency of once every 60 seconds. **Note** A lower timer setting causes CDP updates to be sent more frequently. • In this example, CDP update packets are configured to be sent at a frequency of once every 20 seconds.
Step 5	**commit**	
Step 6	(Optional) **show cdp** **Example:** `RP/0/RP0/CPU0:router# show cdp`	Displays global CDP information. The output displays the CDP version running on the router, the hold time setting, and the timer setting.

Monitoring CDP

This task shows how to monitor CDP.

Note The commands can be entered in any order.

SUMMARY STEPS

1. **show cdp entry** {***** | *entry-name*} [**protocol** | **version**]
2. **show cdp interface** [*type interface-path-id* | **location** *node-id*]
3. **show cdp neighbors** [*type interface-path-id* | **location** *node-id*] [**detail**]
4. **show cdp traffic** [**location** *node-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show cdp entry** {*****	*entry-name*} [**protocol**	**version**] **Example:** `RP/0/RSP0/CPU0:router# show cdp entry *`	Displays information about a specific neighboring device or all neighboring devices discovered using CDP.
Step 2	**show cdp interface** [*type interface-path-id*	**location** *node-id*] **Example:** `RP/0/RSP0/CPU0:router# show cdp interface pos 0/0/0/1`	Displays information about the interfaces on which CDP is enabled.

	Command or Action	Purpose
Step 3	**show cdp neighbors** [*type interface-path-id* | **location** *node-id*] [**detail**] **Example:** `RP/0/RSP0/CPU0:router# show cdp neighbors`	Displays detailed information about neighboring devices discovered using CDP.
Step 4	**show cdp traffic** [**location** *node-id*] **Example:** `RP/0/RSP0/CPU0:router# show cdp traffic`	Displays information about the traffic gathered between devices using CDP.

Examples

The following is sample output for the **show cdp neighbors** command:

```
RP/0/RP0/CPU0:router# show cdp neighbors

  Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                    S - Switch, H - Host, I - IGMP, r - Repeater

  Device ID        Local Intrfce   Holdtme  Capability  Platform  Port ID
  router1          Mg0/0/CPU0/0    177      T S         WS-C2924M Fa0/12
  router2          PO0/4/0/0       157      R           12008/GRP PO0/4/0/1
```

The following is sample output for the **show cdp neighbors** command. In this example, the optional *type instance* arguments are used in conjunction with the **detail** optional keyword to display detailed information about a CDP neighbor. The output includes information on both IPv4 and IPv6 addresses.

```
RP/0/RP0/CPU0:router# show cdp neighbors POS 0/4/0/0 detail

  -------------------------
  Device ID: uut-user
  SysName : uut-user
  Entry address(es):
  IPv4 address: 1.1.1.1
  IPv6 address: 1::1
  IPv6 address: 2::2
  Platform: cisco 12008/GRP, Capabilities: Router
  Interface: POS0/4/0/3
  Port ID (outgoing port): POS0/2/0/3
  Holdtime : 177 sec

  Version :
  Cisco IOS XR Software, Version 0.0.0[Default]
  Copyright (c) 2005 by cisco Systems, Inc.

  advertisement version: 2
```

The following is sample output for the **show cdp entry** command. In this example, the optional *entry* argument is used to display entry information related to a specific CDP neighbor.

```
RP/0/RP0/CPU0:router# show cdp entry router2
```

```
  advertisement version: 2

  -------------------------
  Device ID: router2
  SysName : router2
  Entry address(es):
  Platform: cisco 12008/GRP,  Capabilities: Router
  Interface: POS0/4/0/0
  Port ID (outgoing port): POS0/4/0/1
  Holdtime : 145 sec

  Version :
  Cisco IOS XR Software, Version 0.48.0[Default]
  Copyright (c) 2004 by cisco Systems, Inc.

  advertisement version: 2
```

The following is sample output for the **show cdp interface** command. In this example, CDP information related to Packet over SONET/SDH (POS) interface 0/4/0/0 is displayed.

```
RP/0/RP0/CPU0:router# show cdp interface pos 0/4/0/0

  POS0/4/0/0 is Up
    Encapsulation HDLC
    Sending CDP packets every 60 seconds
    Holdtime is 180 seconds
```

The following is sample output for the **show cdp traffic** command:

```
RP/0/RP0/CPU0:router# show cdp traffic

  CDP counters :
          Packets output: 194, Input: 99
          Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
          No memory: 0, Invalid packet: 0, Truncated: 0
          CDP version 1 advertisements output: 0, Input: 0
          CDP version 2 advertisements output: 194, Input: 99
          Unrecognize Hdr version: 0, File open failed: 0
```

The following is sample output for the **show cdp traffic** command. In this example, the optional **location** keyword and *node-id* argument are used to display information about the traffic gathered between devices using CDP from the specified node.

```
RP/0/RP0/CPU0:router# show cdp traffic location 0/4/cpu0

  CDP counters :
          Packets output: 16, Input: 13
          Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
          No memory: 0, Invalid packet: 0, Truncated: 0
          CDP version 1 advertisements output: 0, Input: 0
          CDP version 2 advertisements output: 16, Input: 13
          Unrecognize Hdr version: 0, File open failed: 0
```

Configuration Examples for Implementing CDP

Enabling CDP: Example

The following example shows how to configure CDP globally and then enable CDP on Packet over SONET/SDH (POS) interface 0/3/0/0:

```
cdp
  interface POS0/3/0/0
   cdp
```

Modifying Global CDP Settings: Example

The following example shows how to modify global CDP settings. In this example, the timer setting is set to 20 seconds, the hold-time setting is set to 30 seconds, and the version of CDP used to communicate with neighboring devices is set to CDPv1:

```
cdp timer 20
  cdp holdtime 30
  cdp advertise v1
```

The following example shows how to use the **show cdp** command to verify the CDP global settings:

```
RP/0/RP0/CPU0:router# show cdp

  Global CDP information:
          Sending CDP packets every 20 seconds
          Sending a holdtime value of 30 seconds
          Sending CDPv2 advertisements is not enabled
```

Additional References

The following sections provide references related to implementing CDP on Cisco IOS XR software.

Related Documents

Related Topic	Document Title
Cisco IOS XR CDP commands	*CDP Commands on Cisco IOS XR Software* module of *System Management Command Reference for isco CRS Routers*
Cisco IOS XR commands	*Cisco IOS XR Commands Master List for the Cisco CRS Router*

Related Topic	Document Title
Cisco IOS XR XML API material	*Cisco IOS XR XML API Guide for the Cisco CRS Router*
Getting started with Cisco IOS XR Software	*Cisco IOS XR Getting Started Guide for the Cisco CRS Router*
Information about user groups and task IDs	*Configuring AAA Services on Cisco IOS XR Software* module of *System Security Configuration Guide for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/cisco/web/support/index.html