Cisco IOS XR Getting Started Guide for the Cisco XR 12000 Series Router

Cisco IOS XR Software Release 4.1

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

Text Part Number: OL-24755-01

Cisco IOS XR Getting Started Guide for the Cisco XR 12000 Series Router

CONTENTS

Preface

This guide describes how to create the initial configuration for a router using the Cisco IOS XR software. This guide also describes how to complete additional administration, maintenance, and troubleshooting tasks that may be required after initial configuration.

This preface contains the following sections:

Changes to This Document

Table 1 lists the technical changes made to this document since it was first printed.

Table 1 Changes to This Document

Revision	Date	Change Summary
OL-24755-01	April 2011	Initial release of this document.

About This Document

The following sections provide information about *Cisco IOS XR Getting Started Guide for the Cisco XR 12000 Series Router* and related documents:

Intended Audience

This document is intended for the following people:

- Experienced service provider administrators
- Cisco telecommunications management engineers
- Third-party field service technicians who have completed the Cisco IOS XR software training sessions
- Customers who daily use and manage routers running Cisco IOS XR software

Organization of the Document

This document contains the following chapters:

- Introduction to Cisco IOS XR Software
- Bringing Up the Cisco IOS XR Software on a Standalone Router
- Configuring General Router Features
- Configuring Additional Router Features
- CLI Tips, Techniques, and Shortcuts
- Troubleshooting the Cisco IOS XR Software
- Understanding Regular Expressions, Special Characters, and Patterns

Related Documents

For a complete listing of available documentation for the Cisco IOS XR software and the routers on which it operates, see the following URLs:

- Cisco IOS XR Software Documentation
  http://www.cisco.com/en/US/products/ps5845/tsd_products_support_series_home.html
  - *Cisco IOS XR ROM Monitor Guide*
    Cisco IOS XR System Management Configuration Guide
    Cisco IOS XR System Security Configuration Guide
    Cisco IOS XR Routing Configuration Guide
    Cisco IOS XR Interface and Hardware Component Configuration Guide
    http://www.cisco.com/en/US/products/ps5845/products_installation_and_configuration_guides_list.html
  - *Cisco IOS XR Interface and Hardware Component Command Reference*
    Cisco IOS XR Routing Command Reference
    http://www.cisco.com/en/US/products/ps5845/prod_command_reference_list.html
- Cisco XR 12000 Series Router Documentation
  http://www.cisco.com/en/US/products/ps6342/tsd_products_support_series_home.html

Note Cisco IOS XR software runs only on the Cisco XR 12000 Series Routers listed in the "Supported Standalone System Configurations" section on page 1-1.

Conventions

This document uses the following conventions:

Convention	Item
boldface font	Commands and keywords
italic font	Variable for which you supply values
`screen` font	Displayed session and system information
`boldface screen` font	Commands and keywords you enter in an interactive environment
`italic screen` font	Variables you enter in an interactive environment
boldface font	Menu items and button names
Option > Network Preferences	Menu navigation

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the publication.

Tip Means *the following information will help you solve a problem.* The information in tips might not be troubleshooting or an action, but contains useful information.

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

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.

CHAPTER 1

Introduction to Cisco IOS XR Software

This chapter introduces the routers that support Cisco IOS XR software. It also introduces router concepts, features, and user interfaces.

Contents

Supported Standalone System Configurations

The Cisco IOS XR software runs on the following standalone systems:

- Cisco XR 12006 Router
- Cisco XR 12008 Router
- Cisco XR 12010 Router
- Cisco XR 12012 Router
- Cisco XR 12016 Router
- Cisco XR 12404 Router
- Cisco XR 12406 Router
- Cisco XR 12410 Router
- Cisco XR 12416 Router

Note Many cards operate in both Cisco XR 12000 Series Routers and in Cisco 12000 Series Internal Routers. For the latest information on which cards are supported by the Cisco IOS XR software in Cisco XR 12000 Series Routers and Cisco 12000 Series Internal Routers, see *Release Notes for Command Cisco IOS XR Software Release 4.0.*

Cisco XR 12000 Series Router Overview

The Cisco XR 12000 Series Router is powered by Cisco IOS XR software, allowing service providers to isolate public and private services through the virtualization of a single router into separate physical and logical partitions. Cisco IOS XR software is a unique self-healing and self-defending operating system designed for always-on operation while scaling capacity and adding new services or features. With distributed processing intelligence and robust quality-of-service (QoS) and multicast mechanisms, the Cisco XR 12000 Series Router allows providers to scale services and customers with performance.

Features and Capabilities

The router is a scalable carrier-class distributed forwarding router, which is designed for redundancy, high security and availability, packaging, power, and other requirements needed by service providers.

The router aggregates triple play Multi-service edge and Ethernet service traffic aggregating these services to 10 Gigabit Ethernet IP, MPLS edge, or core. It support Ethernet, serial (including MLPPP), frame relay and POS interface on the access side and Ethernet or POS interfaces on the core side.

The following sections describe the features and capabilities in detail:

Cisco IOS XR Software

The router runs Cisco IOS XR Software, which offers the following:

- Rich Networking Feature Set—Cisco IOS XR Software represents a continuation of the Cisco networking leadership in helping customers realize the power of their networks and the Internet. It provides unprecedented routing-system scalability, high availability, service isolation, and manageability to meet the mission-critical requirements of next-generation networks.
- Operating system infrastructure protection—Cisco IOS XR Software provides a microkernel architecture that forces all but the most critical functions, such as memory management and thread distribution, outside of the kernel, thereby preventing failures in applications, file systems, and even device drivers from causing widespread service disruption.
- Process and thread protection—Each process, even individual process thread, is executed in its own protected memory space, and communications between processes are accomplished through well-defined, secure, and version-controlled application programming interfaces (APIs), significantly minimizing the effect that any process failure can have on other processes.

- Cisco In-Service Software Upgrade (ISSU)—Cisco IOS XR Software modularity sustains system availability during installation of a software upgrade. ISSUs or hitless software upgrades (HSUs) allow you to upgrade most Cisco router software features without affecting deployed services. You can target particular system components for upgrades based on software packages or composites that group selected features. Cisco preconfigures and tests these packages and composites to help ensure system compatibility.
- Process restart—You can restart critical control-plane processes both manually and automatically in response to a process failure versus restarting the entire operating system. This feature supports the Cisco IOS XR Software goal of continuous system availability and allows for quick recovery from process or protocol failures with minimal disruption to customers or traffic.
- State checkpoint—You can maintain a memory and critical operating state across process restarts to sustain routing adjacencies and signaling state during a Route Switch Processor (RSP) switchover.
- Ethernet virtual connections (EVCs)—Ethernet services are supported using individual EVCs to carry traffic belonging to a specific service type or end user through the network. You can use EVC-based services in conjunction with MPLS-based L2VPNs and native IEEE bridging deployments.
- Flexible VLAN classification—VLAN classification into Ethernet flow points (EFPs) includes single-tagged VLANs, double-tagged VLANs (QinQ and IEEE 802.1ad), contiguous VLAN ranges, and noncontiguous VLAN lists.
- IEEE Bridging—Software supports native bridging based on IEEE 802.1Q, IEEE 802.1ad, IEEE 802.1ah provider backbone bridges (PBB) and QinQ VLAN encapsulation mechanisms on the router.
- IEEE 802.1s Multiple Spanning Tree (MST)—MST extends the IEEE 802.1w Rapid Spanning Tree Protocol (MSTP) to multiple spanning trees, providing rapid convergence and load balancing.
- MST Access Gateway—This feature provides a resilient, fast-convergence mechanism for aggregating and connecting to Ethernet-based access rings.
- Virtual Private LAN Services (VPLS)—VPLS is a class of VPN that supports the connection of multiple sites in a single, bridged domain over a managed IP/MPLS network. It presents an Ethernet interface to customers, simplifying the LAN and WAN boundary for service providers and customers, and enabling rapid and flexible service provisioning because the service bandwidth is not tied to the physical interface. All services in a VPLS appear to be on the same LAN, regardless of location.
- Hierarchical VPLS (H-VPLS)—H-VPLS provides a level of hierarchy at the edge of the VPLS network for increased scale. QinQ access and H-VPLS pseudowire access options are supported.
- Virtual Private WAN Services/Ethernet over MPLS (VPWS/EoMPLS)—EoMPLS transports Ethernet frames across an MPLS core using pseudowires. Individual EFPs or an entire port can be transported over the MPLS backbone using pseudowires to an egress interface or subinterface.
- Pseudowire redundancy—Pseudowire redundancy supports the definition of a backup pseudowire to protect a primary pseudowire that fails.
- Multisegment pseudowire stitching—Multisegment pseudowire stitching is a method for interworking two pseudowires together to form a cross-connect relationship.
- IPv4 Multicast—IPv4 Multicast supports Internet Group Management Protocol Versions 2 and 3 (IGMPv2/v3), Protocol Independent Multicast Source Specific Multicast (SSM) and Sparse Mode (SM), Multicast Source Discovery Protocol (MSDP), and Anycast Rendezvous Point (RP).

- IGMP v2/v3 Snooping—This Layer 2 mechanism efficiently tracks multicast membership on an L2VPN network. Individual IGMP joins are snooped at the VLAN level or pseudowire level, and then it summarizes the results into a single upstream join message. In residential broadband deployments, this feature enables the network to send only channels that are being watched to the downstream users.
- N-Tuple Hashing—A new 7-tuple hash algorithm, which provides better load balancing across equal cost paths, is introduced. This algorithm uses additional Layer 4 information from the Layer 3 packet. For more information on 7-tuple hashing, see the *Cisco IOS XR IP Addresses and Services Configuration Guide for the Cisco XR 12000 Series Router.*
- Link Bundling Phase 3—The link bundle interface is enabled to be used as an edge-facing interface by providing a number of new features, such as ACL, Mac Accounting, IPv6, PIMv6, uRPF, MVPN, L2VPN, BFD, and Unequal BW. For more information on link bundling phase 3, see the *Cisco IOS XR Interface and Hardware Component Configuration Guide for the Cisco XR 12000 Series Router.*

Flexible Ethernet

The router uses Ethernet as its transport mechanism, which offers the following:

- Ethernet virtual connections (EVCs)—Ethernet services are supported using individual EVCs to carry traffic belonging to a specific service type or end user through the network. You can use EVC-based services in conjunction with MPLS-based L2VPNs and native IEEE bridging deployments.
- Flexible VLAN classification—VLAN classification into EFPs includes single-tagged VLANs, double-tagged VLANs (QinQ and IEEE 802.1ad), contiguous VLAN ranges, and noncontiguous VLAN lists.
- IEEE Bridging— The software supports native bridging based on IEEE 802.1Q, IEEE 802.1ad, and QinQ VLAN encapsulation mechanisms on the router.
- IEEE 802.1s Multiple Spanning Tree (MST)—MST extends the MSTP to multiple spanning trees, providing rapid convergence and load balancing.
- MST Access Gateway—This feature provides a resilient, fast-convergence mechanism for aggregating and connecting to Ethernet-based access rings.

L2VPN

The router uses L2VPNs, which offers the following:

- Virtual Private LAN Services (VPLS)—VPLS is a class of VPN that supports the connection of multiple sites in a single, bridged domain over a managed IP/MPLS network. It presents an Ethernet interface to customers, simplifying the LAN and WAN boundary for service providers and customers, and enabling rapid and flexible service provisioning because the service bandwidth is not tied to the physical interface. All services in a VPLS appear to be on the same LAN, regardless of location.
- Hierarchical VPLS (H-VPLS)—H-VPLS provides a level of hierarchy at the edge of the VPLS network for increased scale. QinQ access and H-VPLS pseudowire access options are supported.
- Virtual Private WAN Services/Ethernet over MPLS (VPWS/EoMPLS)—EoMPLS transports Ethernet frames across an MPLS core using pseudowires. Individual EFPs or an entire port can be transported over the MPLS backbone using pseudowires to an egress interface or subinterface.
- Pseudowire redundancy—Pseudowire redundancy supports the definition of a backup pseudowire to protect a primary pseudowire that fails.

- Multisegment pseudowire stitching—This feature is a method used to interwork two pseudowires together to form a cross-connect relationship.

Multicast

The router supports multicast, which offers the following:

- IPv4 Multicast—IPv4 Multicast supports Internet Group Management Protocol Versions 2 and 3 (IGMPv2/v3), Protocol Independent Multicast Source Specific Multicast (SSM) and Sparse Mode (SM), Multicast Source Discovery Protocol (MSDP), and Anycast Rendezvous Point (RP).
- IGMP v2/v3 Snooping—This Layer 2 mechanism efficiently tracks multicast membership on an L2VPN network. Individual IGMP joins are snooped at the VLAN level or pseudowire level, and then it summarizes the results into a single upstream join message. In residential broadband deployments, this feature enables the network to send only channels that are being watched to the downstream users.

OAM

The router supports different types of operations, administration, and maintenance (OAM), which offers the following:

- E-OAM (IEEE 802.3ah)—Ethernet link layer OAM is a vital component of EOAM that provides physical-link OAM to monitor link health and assist in fault isolation. Along with IEEE 802.1ag, Ethernet link layer OAM can be used to assist in rapid link-failure detection and signaling to remote end nodes of a local failure.
- E-OAM (IEEE 802.1ag and ITU-T Y.1731)—Ethernet Connectivity Fault Management (CFM) is a service-level OAM protocol that provides a number of mechanisms for fault management and performance monitoring. This includes procedures for monitoring and verifying the path between multiple end points, via IEEE 802.1 bridges and LANs.
- MPLS OAM—This protocol supports Label Switched Path (LSP) ping, LSP TraceRoute, and Virtual Circuit Connectivity Verification (VCCV). .
- Ethernet SLA (Service Level Agreement)—The router supports a feature-rich manageability interface for performance monitoring, using the capabilities provided by the Ethernet CFM.
- Ethernet Fault Detection—The router supports a mechanism to use faults detected by Ethernet OAM protocols as a trigger to bring down interfaces or sub-interfaces. Hence, there is a trigger protection switching or L3 re-routing during a failure.

Layer 3 Routing

The router runs Cisco IOS XR Software, which supports Layer 3 routing and a range of IPv4 services and routing protocols, including the following:

- Intermediate System-to-Intermediate System (IS-IS)—Integrated Intermediate IS-IS, Internet Protocol Version 4 (IPv4), is a standards-based Interior Gateway Protocol (IGP). For more information on IS-IS, see *Cisco IOS XR Routing Configuration Guide for the Cisco XR 12000 Series Router*.
- Open Shortest Path First (OSPF)—OSPF is an IGP developed by the OSPF working group of the Internet Engineering Task Force (IETF). For more information on OSPF, see *Cisco IOS XR Routing Configuration Guide for the Cisco XR 12000 Series Router*.

- Static Routing—Static routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. For more information on static routing, see *Cisco IOS XR Routing Configuration Guide for the Cisco XR 12000 Series Router*.
- IPv4 Multicast—IPv4 Multicast delivers source traffic to multiple receivers without adding any additional burden on the source or the receivers while using the least network bandwidth of any competing technology. For more information on IPv4 Multicast, see *Cisco IOS XR Multicast Configuration Guide for the Cisco XR 12000 Series Router*.
- Routing Policy Language (RPL)—RPL provides a single, straightforward language in which all routing policy needs can be expressed. For more information on RPL, see *Cisco IOS XR Routing Configuration Guide for the Cisco XR 12000 Series Router*.
- Hot Standby Router Protocol (HSRP)—HSRP is an IP routing redundancy protocol designed to allow for transparent failover at the first-hop IP router. For more information on HSRP, see *Cisco IOS XR IP Addresses and Services Configuration Guide for the Cisco XR 12000 Series Router*.
- Virtual Router Redundancy Protocol (VRRP)—VRRP allows for transparent failover at the first-hop IP router, enabling a group of routers to form a single virtual router. For more information on VRRP, see *Cisco IOS XR IP Addresses and Services Configuration Guide for the Cisco XR 12000 Series Router*.
- Border Gateway Protocol (BGP) Add Path— This feature enables a BGP speaker to send multiple paths for a prefix. For more information on BGP Add Path, see *Cisco IOS XR Routing Configuration Guide for the Cisco XR 12000 Series Router*.
- Selective VRF Download (SVD)—This feature allows the download of only those prefixes and labels to a line card that are actively required to forward traffic through that line card. For more information on SVD, see the *Cisco IOS XR Routing Configuration Guide for the Cisco XR 12000 Series Router*.

MPLS VPN

The router supports MPLS VPN, which offers the following:

- MPLS L3VPN—This IP VPN feature for MPLS allows a Cisco IOS Software or Cisco IOS XR software network to deploy scalable IPv4 Layer 3 VPN backbone services. An IP VPN is the foundation that companies use for deploying or administering value-added services, including applications and data hosting network commerce and telephony services, to business customers.
- Carrier Supporting Carrier (CSC)—CSC allows an MPLS VPN service provider to connect geographically isolated sites using another backbone service provider and still maintain a private address space for its customer VPNs. It is implemented as defined by IETF RFC 4364.
- Inter-AS—is a peer-to-peer type model that allows extension of VPNs through multiple provider or multi-domain networks. This lets service providers peer up with one another to offer end-to-end VPN connectivity over extended geographical locations. An MPLS VPN Inter-AS allows:
  - VPN to cross more than one service provider backbone.
  - VPN to exist in different areas.
  - Confederations to optimize Internal Border Gateway Protocol (iBGP) meshing.

QoS

The router supports many types of quality of service (QoS), which offers the following:

- QoS—Comprehensive QoS support with up to 3 million queues, Class-Based Weighted Fair Queuing (CBWFQ) based on a three-parameter scheduler, Weighted Random Early Detection (WRED), two-level strict priority scheduling with priority propagation, and 2-rate, 3-color (2R3C) Policing are all supported.
- Cisco IOS XR Software—This software supports a rich variety of QoS mechanisms, including policing, marking, queuing, dropping, and shaping. In addition, the operating systems support Modular QoS CLI (MQC). Modular CLI is used to configure various QoS features on various Cisco platforms.
- H-QoS—Is supported on Ethernet interfaces. For EVCs four-level H-QoS support is provided with the following hierarchy levels: port, group of EFPs, EFP, and class of service. This level of support allows for per-service and per-end user QoS granularity. Four-level H-QoS support is provided for EVCs with the following hierarchy levels: port, group of EFPs, EFP, and class of service. This level of support allows for per-service and per-end user QoS granularity. H-QOS support is also provided on SIP based interfaces.

MPLS TE

The router supports MPLE Traffic Engineering (TE), which offers the following:

- MPLS TE—Cisco IOS XR Software supports MPLS protocols such as Traffic Engineering/Fast Reroute (TE-FRR), Resource Reservation Protocol (RSVP), Label Distribution Protocol (LDP), and Targeted Label Distribution Protocol (T-LDP).
- MPLS TE Preferred Path—Preferred tunnel path functions let you map pseudowires to specific TE tunnels. Attachment circuits are cross-connected to specific MPLS TE tunnel interfaces instead of remote provider-edge router IP addresses (reachable using IGP or LDP).
- Ignore Intermediate System-to-Intermediate System (IS-IS) Overload Bit Avoidance—This feature allows network administrators to prevent a RSVP-TE Label Switched Path (LSP) from being disabled when a router in that path has its Intermediate System-to-Intermediate System (IS-IS) overload bit set. For more information on IS-IS overload bit aviodance, see the *Cisco IOS XR MPLS Configuration Guide for the Cisco XR 12000 Series Router.*

Router Management Interfaces

Because new routers are not yet configured for your environment, you must begin the configuration using the command-line interface (CLI). This guide provides instructions on using the CLI to configure basic router features. Cisco IOS XR software supports the following router management interfaces, which are described in the following sections:

- Command-Line Interface, page 1-7
- Extensible Markup Language API, page 1-8
- Simple Network Management Protocol, page 1-8

Command-Line Interface

The CLI is the primary user interface for configuring, monitoring, and maintaining routers that run Cisco IOS XR software. The CLI allows you to directly and simply execute Cisco IOS XR commands.

All procedures in this guide use CLI. Before you can use other router management interfaces, you must first use the CLI to install and configure those interfaces. Guidelines for using the CLI to configure the router are discussed in the following chapters:

- Configuring General Router Features
- Configuring Additional Router Features
- CLI Tips, Techniques, and Shortcuts

For more information on CLI procedures for other tasks, such as hardware interface and software protocol management tasks, see the Cisco IOS XR software documents listed in the "Related Documents" section on page x.

Extensible Markup Language API

The Extensible Markup Language (XML) application programming interface (API) is an XML interface used for rapid development of client applications and perl scripts to manage and monitor the router. Client applications can be used to configure the router or request status information from the router by encoding a request in XML API tags and sending it to the router. The router processes the request and sends the response to the client in the form of encoded XML API tags. The XML API supports readily available transport layers, including Telnet, SSH, and Common Object Request Broker Architecture (CORBA). The Secure Socket Layer (SSL) transport is also supported by the XML API.

For more information, see the Cisco IOS XR software documents listed in the "Related Documents" section on page x.

Simple Network Management Protocol

Simple Network Management Protocol (SNMP) is an application-layer protocol that facilitates the exchange of management information between network devices. By using SNMP-transported data (such as packets per second and network error rates), network administrators can manage network performance, find and solve network problems, and plan for network growth.

The Cisco IOS XR software supports SNMP v1, v2c, and v3. SNMP is part of a larger architecture called the Internet Network Management Framework (NMF), which is defined in Internet documents called RFCs. The SNMPv1 NMF is defined by RFCs 1155, 1157, and 1212, and the SNMPv2 NMF is defined by RFCs 1441 through 1452..

SNMP is a popular protocol for managing diverse commercial internetworks and those used in universities and research organizations. SNMP-related standardization activity continues even as vendors develop and release state-of-the-art, SNMP-based management applications. SNMP is a relatively simple protocol, yet its feature set is sufficiently powerful to handle the difficult problems presented in trying to manage the heterogeneous networks of today.

For more information, see the Cisco IOS XR software documents listed in the "Related Documents" section on page x.

Selecting and Identifying the Designated Shelf Controller

The designated shelf controller (DSC) controls a standalone router or a multishelf system. A DSC is a role that is assigned to one performance route processor (PRP) card in each router . A DSC is a role that is assigned to one performance route processor (PRP) card in each router or multishelf system. PRP cards operate in Cisco XR 12000 and 12000 Series Routers.

Note Throughout this guide, the term PRP is used to refer to the PRP cards supported on Cisco XR 12000 Series Routers. Cisco XR 12000 Series Routers, support both the PRP-2 and the PRP-3 cards. If a feature or an issue applies to only one platform, the accompanying text specifies the platform.

Although each router can have two RP cards, only one can serve as the DSC and control the router. The DSC provides system-wide administrative functions, including:

- User configuration using a terminal connection or network connection
- Distribution of software to each node in the router or system
- Coordination of software versioning and configurations for all nodes in the router or system
- Hardware inventory and environmental monitoring

The first step in setting up a new router is to select or identify the DSC because the initial router configuration takes place through the DSC. The following sections describe how to select and identify the DSC on different routers:

- Selecting and Identifying the DSC on Cisco XR 12000 and 12000 Series Routers, page 1-9
- Verifying the DSC, page 1-9

Selecting and Identifying the DSC on Cisco XR 12000 and 12000 Series Routers

A Cisco XR 12000 Series Router or Cisco 12000 Series Internal Router supports multiple PRPs. When the router is started for the first time, the PRP in the slot with the lowest number becomes the active PRP and is identified by the alphanumeric display: ACTV RP. The active PRP serves as the DSC. If another PRP is configured as a standby PRP for the DSC, that PRP can assume the DSC role if the DSC fails.

To have a PRP in a higher-numbered slot become the DSC, you must bring up the router with only that PRP installed. After the chosen PRP becomes the DSC, it remains the DSC after subsequent restarts and you can add the other PRPs.

Note Additional PRPs can be installed to host secure domain routers (SDRs), which are introduced in Chapter 3, "Configuring General Router Features." To configure general router features, you must connect to the DSC. To configure SDR features, you must connect to the PRP for the appropriate SDR.

Verifying the DSC

Use the **show dsc** command to verify which RP is acting as the primary DSC for the router or routing system.
The following example shows sample output of the **show dsc** command on a Cisco XR 12000 Series Router:

```
RP/0/0/CPU0:router#admin
Mon May 31 01:38:09.733 DST
RP/0/0/CPU0:router(admin)#show dsc all
Mon May 31 01:38:31.134 DST

NODE            ROLE        PRIORITY      TBEACON   PRESENT  SERIAL ID
==========================================================================
0/0/CPU0        DSC         3             2000      YES      invalid
--------------------------------------------------------------------------
```

Connecting to the Router Through the Console Port

The first time you connect to a new router with Cisco IOS XR software, you must connect through the Console port on the DSC. Although typical router configuration and management take place using an Ethernet port on the DSC, you must configure the console port for your LAN before it can be used.

Figure 1-1 shows the PRP-2 connections on the Cisco XR 12000 Series Router. Figure 1-2 shows the PRP-3 connections on the Cisco XR 12000 Series Router.

Note Cisco IOS XR software does not support PRP-1.

Figure 1-1 Communication Ports on the PRP-2 for a Cisco XR 12000 Series Router

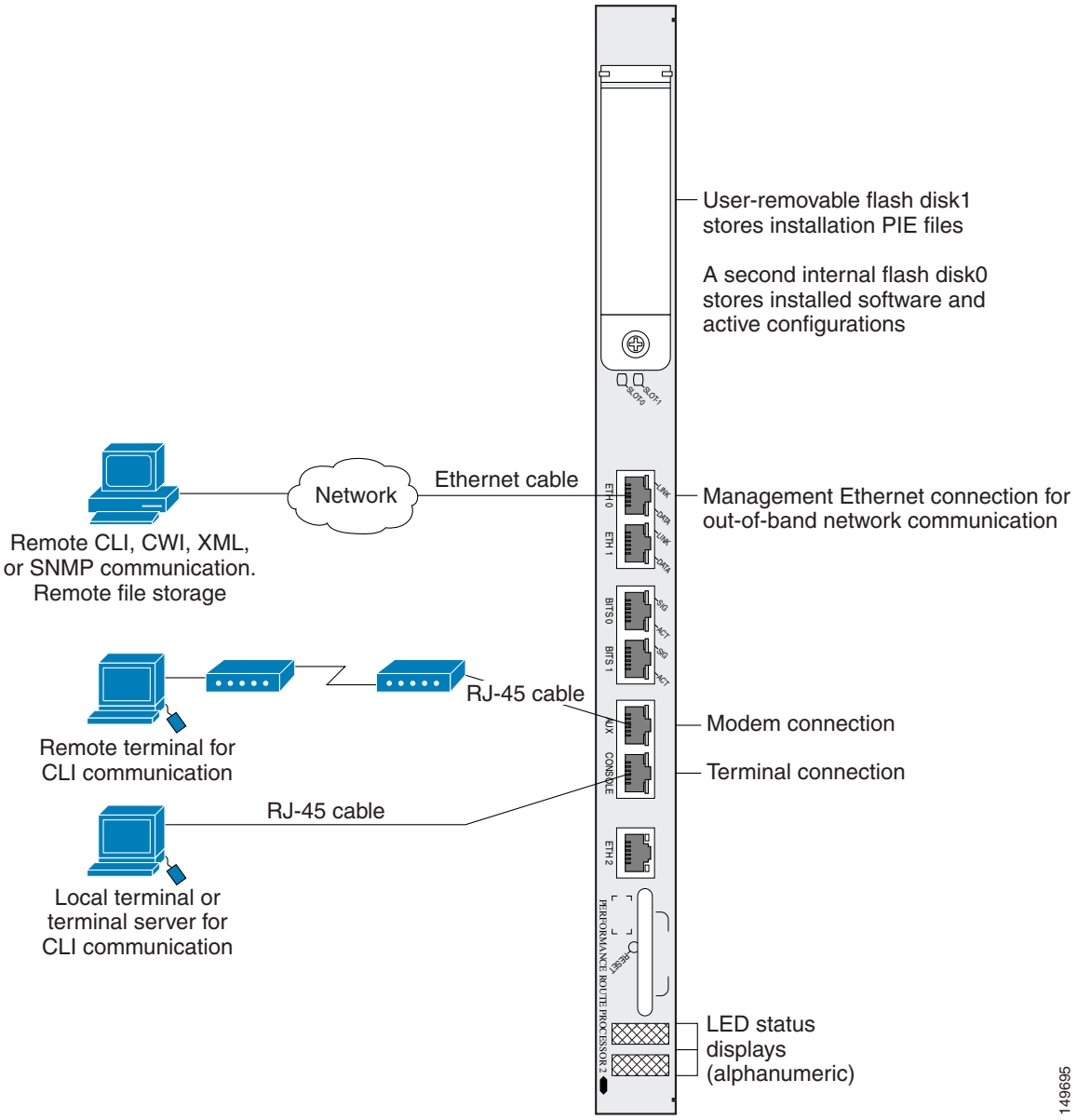
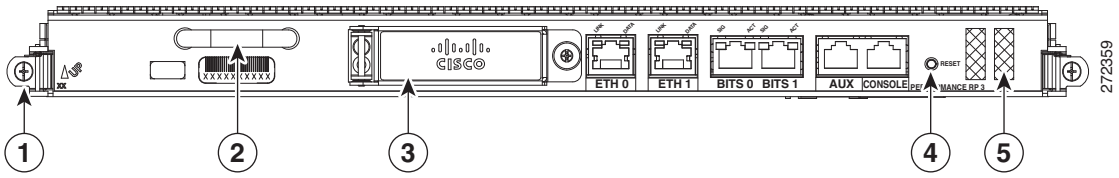


Figure 1-2 Communication Ports on the PRP-3



1	Ejector Lever
2	Handle
3	External Compact Flash
4	Reset button
5	Alphanumeric LEDs

To connect to the router through the Console port, perform the following procedure.

SUMMARY STEPS

1. Power on the standalone router, or power on Rack 0 in a multishelf system.
2. Identify the DSC.
3. Connect a terminal to the Console port of the DSC.
4. Start the terminal emulation program.
5. Press **Enter**.
6. Log in to the router.
7. **admin**
8. **show dsc all**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Power on the standalone router, or power on Rack 0 in a multishelf system.	Starts the router or Rack 0. • This step is required only if the power is not on. • For information on power installation and controls, see the hardware documentation listed in the "Related Documents" section on page x.
Step 2	Identify the DSC.	Identifies the RP to which you must connect in the next step. • For more information, see the "Selecting and Identifying the Designated Shelf Controller" section on page 1-9.
Step 3	Connect a terminal to the Console port of the DSC.	Establishes a communications path to the router. • During the initial setup, you can communicate with the router only through the Console port of the DSC. • Router Console port is designed for a serial cable connection to a terminal or a computer that is running a terminal emulation program. • Terminal settings are: – Bits per second: 9600/9600 – Data bits: 8 – Parity: None – Stop bit: 2 – Flow control: None • For information on the cable requirements for the Console port, see the hardware documentation listed in the "Related Documents" section on page x.

	Command or Action	Purpose
Step 4	Start the terminal emulation program.	(Optional) Prepares a computer for router communications. • Not required if you are connecting through a terminal. • Terminals send keystrokes to, and receive characters, from another device. If you connect a computer to the Console port, you must use a terminal emulation program to communicate with the router. For instructions on using the terminal emulation program, see the documentation for that program.
Step 5	Press **Enter**.	Initiates communication with the router. • If no text or router prompt appears when you connect to the console port, press **Enter** to initiate communications. • If no text appears when you press **Enter**, give the router more time to complete the initial boot procedure, then press **Enter**. • If the prompt gets lost among display messages, press **Enter** again. • If the router has no configuration, the router displays the prompt: `Enter root-system username:` • If the router has been configured, the router displays the prompt: `Username:`
Step 6	Log in to the router.	Establishes your access rights for the router management session. • Enter the root-system username and password or the username and password provided by your system administrator. • After you log in, the router displays the CLI prompt, which is described in the "CLI Prompt" section on page 3-37. • If the router prompts you to enter a root-system username, the router is not configured, and you should follow one of the bring up procedures mentioned in the next section.
Step 7	**admin** **Example:** `RP/0/0/CPU0:router# admin`	Places the router in administration EXEC mode.
Step 8	**show dsc all** **Example:** `RP/0/0/CPU0:router(admin)# show dsc all`	Displays the DSC information for the router or router system so that you can verify that you have connected to the DSC console port.

Where to Go Next

If you have logged into the router or multishelf system, you can perform the general router configuration as described in Configuring General Router Features.

If the router is prompting you to enter a root-system username, bring up the router. For more information, see Bringing Up the Cisco IOS XR Software on a Standalone Router.

CHAPTER 2

Bringing Up the Cisco IOS XR Software on a Standalone Router

This chapter provides instructions for bringing up the Cisco IOS XR software on a standalone router for the first time. This section applies to standalone routers that are delivered with Cisco IOS XR software installed.

Note If you are upgrading a Cisco XR 12000 Series Router from Cisco IOS software to Cisco IOS XR software, see the Cisco IOS XR software document *Upgrading from Cisco IOS to Cisco IOS XR Software on the Cisco 12000 Series Router.*

Contents

Prerequisites

The following sections describe the software and hardware requirements for bringing up a standalone system running Cisco IOS XR Software Release 4.1.

Software Requirements

The system requires compatible ROM Monitor firmware on all RPs.

Caution The ROM Monitor firmware on all RPs must be compatible with the Cisco IOS XR software release currently running on the router before a Cisco XR 12000 Series Router system is upgraded to Cisco IOS XR Software Release 4.1. For minimum ROM Monitor requirements for Cisco IOS XR Software Release 3.2.0 and later releases, see the Software/Firmware Compatibility Matrix at the following URL: http://www.cisco.com/web/Cisco_IOS_XR_Software/index.html

If the router is brought up with an incompatible version of the ROM Monitor software, the standby RP may fail to boot. For instructions to overcome a boot block in the standby RP in a single-chassis system, see *Cisco IOS XR ROM Monitor Guide for the Cisco XR 12000 Series Router*. If a boot block occurs in a multishelf system, contact your Cisco Technical Support representative for assistance. See the "Related Documents" section on page x.

Hardware Prerequisites and Documentation

The Cisco IOS XR software runs on the routers listed in the "Supported Standalone System Configurations" section on page 1-1. Before a router can be started, the following hardware management procedures must be completed:

- Site preparation
- Equipment unpacking
- Router installation

For information on how to complete these procedures for your router equipment, see the hardware documents listed in the "Related Documents" section on page x.

Note If you are upgrading a Cisco 12000 Series Router from Cisco IOS software to Cisco IOS XR software, you must first prepare the router. For more information, see *Upgrading from Cisco IOS to Cisco IOS XR Software on the Cisco 12000 Series Router*. For a complete listing of available documents, see the "Related Documents" section on page x.

Bringing Up and Configuring a Standalone Router

To bring up a standalone router, connect to the router and configure the root-system username and password, as described in the following procedure:

SUMMARY STEPS

1. Establish a connection to the DSC Console port.
2. Type the username for the root-system login and press **Enter**.
3. Type the password for the root-system login and press **Enter.**
4. Log in to the router.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Establish a connection to the DSC Console port.	Initiates communication with the router. • For instructions on connecting to the Console port, see the "Connecting to the Router Through the Console Port" section on page 1-10. • After you have successfully connected to the router through the Console port, the router displays the prompt: `Username:` • If the Username prompt appears, skip this procedure and continue the general router configuration as described in Chapter 4, "Configuring Additional Router Features."
Step 2	Type the username for the root-system login and press **Enter**.	Sets the root-system username, which is used to log in to the router.
Step 3	Type the password for the root-system login and press **Enter**.	Creates an encrypted password for the root-system username. **Note** This password can be changed with the **secret** command.
Step 4	Retype the password for the root-system login and press **Enter**.	Allows the router to verify that you have entered the same password both times. • If the passwords do not match, the router prompts you to repeat the process.
Step 5	Log in to the router.	Establishes your access rights for the router management session. • Enter the root-system username and password that were created earlier in this procedure. • After you log in, the router displays the CLI prompt, which is described in the CLI Prompt.

Examples

The following example shows the root-system username and password configuration for a new router, and it shows the initial log in:

```
--- Administrative User Dialog ---
Enter root-system username: username1
  Enter secret:
  Enter secret again:
RP/0/0/CPU0:Jan 10 12:50:53.105 : exec[65652]: %MGBL-CONFIG-6-DB_COMMIT :
'Administration configuration committed by system'. Use 'show configuration
commit changes 2000000009' to view the changes.
Use the 'admin' mode 'configure' command to modify this configuration.
User Access Verification
Username: username1
Password:
RP/0/0/CPU0:router#
```

The *secret* line in the configuration command script shows that the password is encrypted. When you enter the password during configuration and login, the password is hidden.

Verifying the System After Initial Boot

To verify the status of the router, perform the following procedure:

SUMMARY STEPS

1. **show version**
2. **admin**
3. **show platform** [*node-id*]
4. **exit**
5. **show redundancy**
6. **show environment**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show version** **Example:** `RP/0/0/CPU0:router# show version`	Displays information about the router, including image names, uptime, and other system information.
Step 2	**admin** **Example:** `RP/0/0/CPU0:router# admin`	Places the router in administration EXEC mode.
Step 3	**show platform** [*node-id*] **Example:** `RP/0/0/CPU0:router(admin)# show platform`	Displays information about the status of cards and modules installed in the router. • Some cards support a CPU module and service processor (SP) module. Other cards support only a single module. • A card module is also called a *node*. When a node is working properly, the status of the node in the State column is IOS XR RUN. The status of the supported SPA interface is OK. • The **show platform** *node-id* command is used to display information for a specific node. Replace *node-id* with a node name from the **show platform** command Node column. **Note** To view the status of all the cards and modules, the **show platform** command must be executed in administration EXEC mode.
Step 4	**exit** **Example:** `RP/0/0/CPU0:router(admin)# exit`	Exits the administration EXEC mode.
Step 5	**show redundancy** **Example:** `RP/0/0/CPU0:router# show redundancy`	Displays the state of the primary (active) and standby (inactive) RPs, including the ability of the standby to take control of the system. • If both RPs are working correctly, one node displays active role, the Partner node row displays standby role, and the Standby node row displays Ready.
Step 6	**show environment** **Example:** `RP/0/0/CPU0:router# show environment`	Displays information about the hardware attributes and status.

Examples of show Commands

The following sections provide examples of **show** commands:

show version Command: Example

The following example shows how to display basic information about the router configuration by entering the **show version** command in EXEC mode:

```
RP/0/0/CPU0:router# show version
Mon May 31 02:03:29.133 DST

Cisco IOS XR Software, Version 4.1.0[Default]
Copyright (c) 2010 by Cisco Systems, Inc.

ROM: System Bootstrap, Version 12.0(20090226:235859) [rtauro-sw30346-33S 1.23dev
(0.35)] DEVELOPMENT SOFTWARE
Copyright (c) 1994-2009 by cisco Systems,  Inc.

router uptime is 2 weeks, 6 days, 9 hours, 16 minutes
System image file is "disk0:c12k-os-mbi-4.1.0/mbiprp-rp.vm"

cisco 12406/PRP (7457) processor with 2097152K bytes of memory.
7457 processor at 1266Mhz, Revision 1.2

1 Cisco 12000 Series Performance Route Processor
1 Cisco 12000 Series - Multi-Service Blade Controller
1 Cisco 12000 4-Port ISE ATM Over SONET OC3/STM-1 Controller (4 ATM)
1 Cisco 12000 Series SPA Interface Processor-601/501/401
1 Cisco 12000 Series SPA Interface Processor-600
3 Management Ethernet
6 PLIM_QOS
1 MgmtMultilink
5 SONET/SDH
2 T3
3 Multilink network interface(s)
28 T1
21 E1
21 Serial network interface(s)
5 GigabitEthernet/IEEE 802.3 interface(s)
4 Asynchronous Transfer Mode
1018k bytes of non-volatile configuration memory.
3623092k bytes of disk0: (Sector size 512 bytes).
3623092k bytes of disk1: (Sector size 512 bytes).
65536k bytes of Flash internal SIMM (Sector size 256k).

Configuration register on node 0/0/CPU0 is 0x102
Boot device on node 0/0/CPU0 is disk0:
Package active on node 0/0/CPU0:
c12k-fwding, V 4.1.0[DT_IMAGE], Cisco Systems, at disk0:c12k-fwding-4.1.0.15
I
    Built on Thu May  6 17:07:57 DST 2010
    By sjc-lds-364 in /auto/ioxbuild6/production/4.1.0.DT_IMAGE/c12k/workspa
ce for pie

c12k-doc, V 4.1.0[DT_IMAGE], Cisco Systems, at disk0:c12k-doc-4.1.0
```

```
Built on Thu May  6 20:36:06 DST 2010
```

show platform Command: Example

The **show platform** command displays information on router resources. In EXEC mode, the **show platform** command displays the resources assigned to the secure domain router (SDR) that you are managing. In administration EXEC mode, the **show platform** command displays all router resources.

Note Secure Domain Routers (SDRs) are introduced in Chapter 3, "Configuring General Router Features."

The following EXEC mode example displays the nodes assigned to the default SDR, which is called the *owner SDR*:

```
RP/0/0/CPU0:router# show platform
Mon May 31 02:15:07.484 DST
Node            Type            PLIM            State           Config State
--------------------------------------------------------------------------------
0/0/CPU0        PRP(Active)     N/A             IOS XR RUN      PWR,NSHUT,MON
0/1/CPU0        L3 Service Eng  N/A             Admin Down      PWR,SHUT,MON
0/2/CPU0        L3LC Eng 3      OC3-ATM-4       IOS XR RUN      PWR,NSHUT,MON
0/3/CPU0        L3LC Eng 5+     Jacket Card     IOS XR RUN      PWR,NSHUT,MON
0/3/1           SPA             SPA-IPSEC-2G-2  READY           PWR,NSHUT
0/3/2           SPA             SPA-1XCHSTM1/OC READY           PWR,NSHUT
0/4/CPU0        L3LC Eng 5      Jacket Card     IOS XR RUN      PWR,NSHUT,MON
0/4/0           SPA             SPA-5X1GE       READY           PWR,NSHUT
```

The following administration EXEC mode example shows all router nodes:

```
RP/0/0/CPU0:router# admin
RP/0/0/CPU0:router(admin)# show platform
Mon May 31 02:18:17.048 DST
Node            Type            PLIM            State           Config State
--------------------------------------------------------------------------------
0/0/CPU0        PRP(Active)     N/A             IOS XR RUN      PWR,NSHUT,MON
0/1/CPU0        L3 Service Eng  N/A             Admin Down      PWR,SHUT,MON
0/2/CPU0        L3LC Eng 3      OC3-ATM-4       IOS XR RUN      PWR,NSHUT,MON
0/3/CPU0        L3LC Eng 5+     Jacket Card     IOS XR RUN      PWR,NSHUT,MON
0/3/1           SPA             SPA-IPSEC-2G-2  READY           PWR,NSHUT
0/3/2           SPA             SPA-1XCHSTM1/OC READY           PWR,NSHUT
0/4/CPU0        L3LC Eng 5      Jacket Card     IOS XR RUN      PWR,NSHUT,MON
0/4/0           SPA             SPA-5X1GE       READY           PWR,NSHUT
0/17/CPU0       CSC6(P)         N/A             PWD             PWR,NSHUT,MON
0/18/CPU0       SFC6            N/A             PWD             PWR,NSHUT,MON
0/19/CPU0       SFC6            N/A             PWD             PWR,NSHUT,MON
0/20/CPU0       SFC6            N/A             PWD             PWR,NSHUT,MON
0/24/CPU0       ALARM6          N/A             PWD             PWR,NSHUT,MON
0/25/CPU0       ALARM6          N/A             PWD             PWR,NSHUT,MON
0/28/CPU0       GSR6-BLOWER     N/A             PWD             PWR,NSHUT,MON
RP/0/0/CPU0:router# end
```

Note LCs in Cisco CRS routers are called modular services cards (MSCs).

For more information on node IDs, see *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router.*

For more information on the **show platform** command, see *Cisco IOS XR Interface and Hardware Component Command Reference for the Cisco XR 12000 Series Router.*

show redundancy Command: Example

The following example shows how to display information about the active and standby (inactive) RPs by entering the **show redundancy** command:

```
RP/0/0/CPU0:router# show redundancy

Mon May 31 02:22:27.482 DST
Redundancy information for node 0/0/CPU0:
==========================================
Node 0/0/CPU0 is in ACTIVE role
Node 0/0/CPU0 has no valid partner

Reload and boot info
----------------------
PRP reloaded Mon May 10 16:47:10 2010: 2 weeks, 6 days, 9 hours, 35 minutes ago
Active node booted Mon May 10 16:47:10 2010: 2 weeks, 6 days, 9 hours, 35 minutes ago

Active node reload "Cause: Turboboot completed successfully"
```

show environment Command: Example

To display environmental monitor parameters for the system, use the **show environment** command in EXEC or administration EXEC mode. The **show environment** [options] command syntax is used.

Enter the **show environment ?** command to display the command options:

```
RP/0/0/CPU0:router# show environment temperatures
```

The following example shows temperature information for a Cisco XR 12000 Series Router:

```
Mon May 31 02:27:06.397 DST
R/S/I   Modules Sensor          Temp. (deg C)

0/0/*     host  Inlet              32.5
          host  Hot                34.0
0/2/*     host  Inlet              31.0
          host  Hot                39.0
0/3/*     host  Inlet              32.0
          host  Hot                42.0
          spa2  Chan4              50.0
          spa2  Chan3              43.0
          spa2  Chan2              48.0
          spa2  Chan1              39.0
          spa2  Local              48.0
          spa1  Chan4              37.0
          spa1  Chan3              33.0
          spa1  Chan2              39.0
          spa1  Chan1              33.0
          spa1  Local              35.0
0/4/*     host  Hot                38.5
          host  Inlet              36.5
          spa0  DownStream_        35.0
          spa0  UpStream_          31.5
```

For more information, see *Cisco IOS XR Interface and Hardware Component Command Reference for the Cisco XR 12000 Series Router*.

Where to Go Next

For information on configuring basic router features, see Configuring General Router Features.

CHAPTER 3

Configuring General Router Features

This chapter describes how to communicate with the router using the command-line interface (CLI), and it also shows basic Cisco IOS XR software configuration management.

Contents

Secure Domain Routers

Cisco XR 12000 Series Routers can be partitioned into multiple, independent routers known as *secure domain routers* (SDRs). Every router is shipped with a default SDR, which is called the *owner SDR*, by default, owns all RPs and Line Cards (LCs) installed in the routing system. To build additional SDRs, you must perform the following steps:

- Create each SDR using configuration commands
- Name the SDR
- Assign RP, PRP and LCs to the SDR
- Configure the interfaces on the LCs on the new SDR

An SDR is a group of cards within a router that is configured to operate as an independent router. SDRs that are created with configuration commands are called SDRs and are configured with custom names to distinguish them from the owner SDR and other named SDRs.

Note In previous releases, SDRs were called logical routers (LRs).

SDRs perform routing functions in the same manner as a physical router, but share some chassis resources with the rest of the system. For example, the applications, configurations, protocols, and routing tables assigned to an SDR belong to that SDR only, but other functions, such as chassis control, switch fabric, and partitioning, are shared with the rest of the system.

To manage the owner SDR, you must connect to the active RP for the owner SDR. In administration configuration mode, you can define new SDRs and assign resources to them (such as DRPs, MSCs, and line cards). In configuration mode, you can configure the operation of the owner SDR. Although you can reassign cards from one SDR to another, you cannot configure and manage cards assigned to a named SDR. To manage cards assigned to a named SDR, you must connect to the appropriate named SDR.

When you manage a named SDR, you must connect to the active RP for that named SDR. You can connect to the named SDR using any of the connection methods you use for the owner SDR (for example, you can connect through the console port or the Management Ethernet interface), and you have control over only the cards assigned to that named SDR. For example, you cannot configure and manage interfaces on LCs assigned to the owner SDR or other SDRs unless you connect directly to those SDRs.

Note Cisco IOS XR Software Release 3.2 supports multiple SDRs on Cisco XR 12000 Series Routers. Cisco IOS XR Software Release 3.3 and later releases support multiple SDRs on the Cisco CRS routers and Cisco XR 12000 Series Routers. For more information, see *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router.*

Connecting and Communicating with the Router

To manage or configure a router running Cisco IOS XR software, you must first connect to the router using a terminal or a PC. Before you connect to the router, you must determine which router entity to manage. You can manage the following router entities:

- Owner SDR. Connect to the designated shelf controller (DSC).
- Router or multishelf system hardware. Connect to the DSC.
- Named SDR. For Cisco XR 12000 Series Routers, connect to the RP that serves as the DSDRSC for that named SDR.

Connections are made either through a direct physical connection to the console port of the DSC or DSDRSC or from a remote location using a modem or an Ethernet connection that leads to the DSC or DSDRSC.

Figure 3-1 shows the DRP PLIM connections.

Figure 3-2 shows the performance route processor 2 (PRP-2) connections for a Cisco XR 12000 Series Router.

Figure 3-3 shows the performance route processor 3(PRP-3) connections for a Cisco XR 12000 Series Router.

The first time a router is started, you must use a direct connection to the DSC Console port to connect to the router and enter the initial configuration information. When the router is directly connected to the Console port, enter CLI commands at a terminal or at a computer running terminal emulation software. This direct Console port connection is useful for entering initial configurations and performing some debugging tasks.

This chapter describes some of the tasks to perform during your initial configuration. One of those tasks is the configuration of the Management Ethernet interface, which is described in the "Configuring the Management Ethernet Interface" section on page 3-66. After the Management Ethernet interface is configured, most router management and configuration sessions take place over an Ethernet network connected to the Management Ethernet interface. SNMP agents also use the network connection.

You can use the modem connection for remote communications with the router. If the Management Ethernet interface fails, the modem connection serves as the alternate remote communications path.

Figure 3-1 Communication Ports on the DRP PLIM

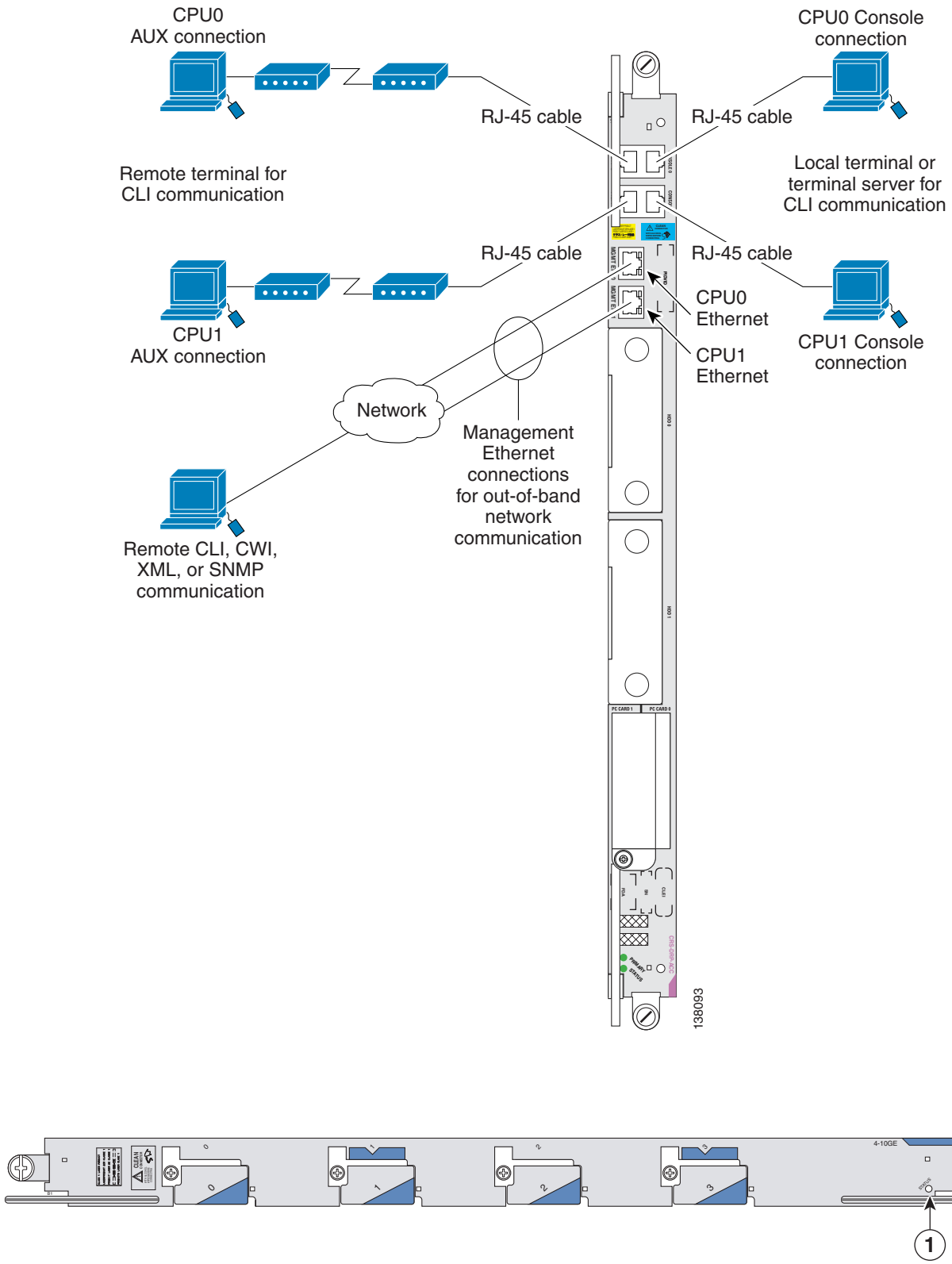


Figure 3-2 Communication Ports on the PRP-2 for a Cisco XR 12000 Series Router

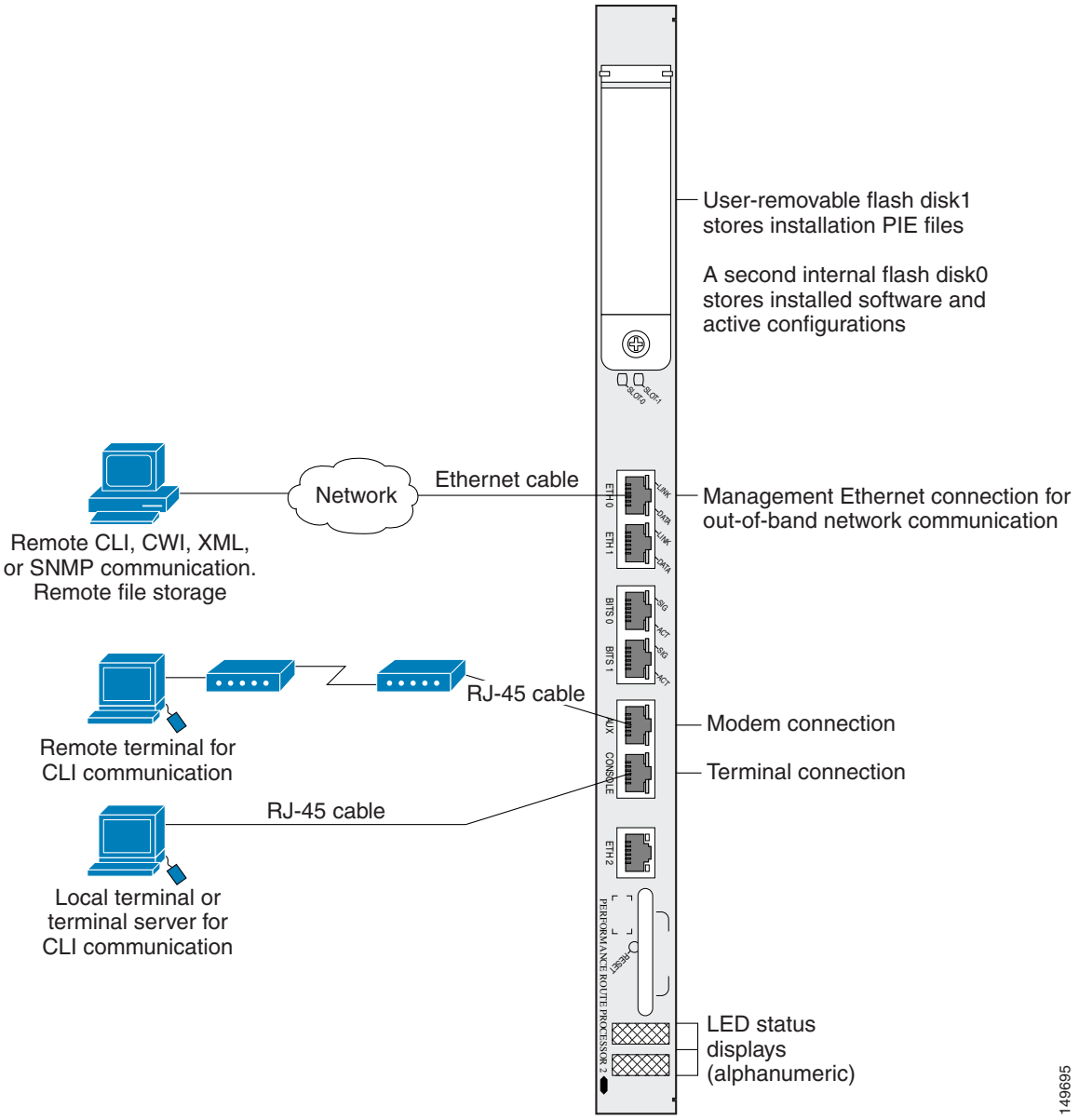
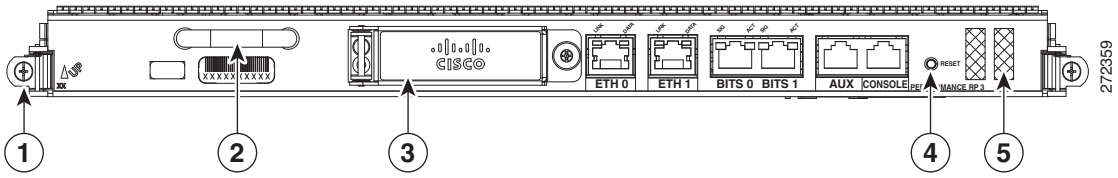


Figure 3-3 Communication Ports on the PRP-3



1	Ejector Lever
2	Handle
3	External Compact Flash
4	Reset button
5	Alphanumeric LEDs

The following sections describe three ways to connect to the router:

- Establishing a Connection Through the Console Port, page 3-32
- Establishing a Connection Through a Terminal Server, page 3-34
- Establishing a Connection Through the Management Ethernet Interface, page 3-36

Establishing a Connection Through the Console Port

To connect to the router through the console port, perform the following procedure.

SUMMARY STEPS

1. Identify the active RP or DRP.
2. Connect a terminal to the Console port of the active RP or DRP.
3. Start the terminal emulation program.
4. Press **Enter**.
5. Log in to the router.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Identify the active RP or DRP.	Identifies the RP or DRP to which you must connect in the next step. • This step is not required when the router hosts only one RP. • On a Cisco XR 12000 Series Router, the active RP is identified by the alphanumeric display: ACTV RP. •
Step 2	Connect a terminal to the Console port of the active RP or DRP.	Establishes a communications path to the router. • During the initial setup, you can communicate with the router only through the console port of the active RP. • Router console port is designed for a serial cable connection to a terminal or a computer that is running a terminal emulation program. • Terminal settings are: – Bits per second: 9600 (default value) – Data bits: 8 – Parity: None – Stop bit: 2 – Flow control: None • For information on the cable requirements for the console port, see the hardware documentation listed in the "Related Documents" section on page x.
Step 3	Start the terminal emulation program.	(Optional) Prepares a computer for router communications. • This step is not required if you are connecting through a terminal. • Terminals send keystrokes to, and receive characters from, another device. If you connect a computer to the Console port, you must use a terminal emulation program to communicate with the router. For instructions on using a terminal emulation program, see the hardware documentation listed in the "Related Documents" section on page x.

	Command or Action	Purpose
Step 4	Press **Enter**.	Initiates communication with the router. • If no text or router prompt appears when you connect to the Console port, press **Enter** to initiate communications. • If no text appears when you press **Enter** and the router has been started recently, give the router more time to complete the initial boot procedure, then press **Enter**. • If the router has no configuration, the router displays the prompt: `Enter root-system username:`. For more information on when a standalone router is starting up for the first time, see Chapter 2, "Bringing Up the Cisco IOS XR Software on a Standalone Router." • If the router has been configured, the router displays the prompt: `Username:`
Step 5	Log in to the router.	Establishes your access rights for the router management session. • Enter the username and password, as described in the "Logging In to a Router or an SDR" section on page 3-36. • After you log in, the router displays the CLI prompt, which is described in the "CLI Prompt" section on page 3-37.

Establishing a Connection Through a Terminal Server

A terminal server connection provides a way to access the Console port from a remote location. It is less expensive to connect to the router through the Management Ethernet interface (because you do not have the additional cost of a terminal server). However, if you need to perform tasks that require Console port access from a remote location, a terminal server is the best method.

The procedure for connecting to the router through a terminal server is similar to the procedure for directly connecting through the Console port. For both connection types, the physical connection takes place through the Console port. The difference is that the terminal server connects directly to the Console port, and you must use a Telnet session to establish communications through the terminal server to the router.

To establish a connection through a terminal server, perform the following procedure.

SUMMARY STEPS

1. Install and configure the terminal server.
2. Connect the terminal server to the Console port of the target RP or DRP.
3. Power on the router.
4. Identify the target RP or DRP.
5. **telnet** *access-server-address port*
6. Press **Enter**.
7. Log in to the router.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Install and configure the terminal server.	Prepares the terminal server for communications with the router and with Telnet clients. • This step is usually preformed once. • For router access, users need the Telnet server IP address and port number for each RP they access. • For additional information on configuring terminal services, including terminal servers and templates, see *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router*.
Step 2	Connect the terminal server to the Console port of the target RP or DRP.	Establishes a communications path between the terminal server and the router. • During the initial router setup, you can communicate with the router only through the Console port of the primary RP. • The router Console port is designed for a serial cable connection to a terminal or terminal server. • The terminal settings are: – Bits per second: 9600(default value) – Data bits: 8 – Parity: None – Stop bit: 2 – Flow control: None • For information on the cable requirements for the Console port, see the hardware documentation listed in the "Related Documents" section on page x. •
Step 3	Power on the router.	Starts the router. • This step is required only if the router power is not on. • For information on power installation and controls, see the hardware documentation listed in the "Related Documents" section on page x.
Step 4	Identify the target RP or DRP.	Identifies the RP or DRP to which you connect in the next step. • This step is not required when the router hosts only one RP or DRP. • On a Cisco XR 12000 Series Router, the active RP is identified by the alphanumeric display: ACTV RP. • If you cannot see the RPs, use a Management Ethernet interface connection to determine which RP is active, or establish terminal server connections to both RPs and then try both.
Step 5	**telnet** *access-server-address port*	Establishes a Telnet session with the terminal server. • Replace *access-server-address* with the IP address of the terminal server, and replace *port* with the terminal server port number that connects to the target RP Console port.

	Command or Action	Purpose
Step 6	Press **Enter**.	(Optional) Initiates communications with the RP or DRP. • If no text or router prompt appears when you start the Telnet session, press **Enter** to initiate communications. • If the router has no configuration, the router displays the prompt: `Enter root-system username:` Enter the root-system username and password when prompted. • If the router has been configured, the router displays the prompt: `Username:`
Step 7	Log in to the router.	Establishes your access rights for the router management session. • Enter a username and password when prompted.

Establishing a Connection Through the Management Ethernet Interface

The Management Ethernet interface allows you to manage the router using a network connection. Before you can use the Management Ethernet interface, the interface must be configured as described in the "Configuring the Management Ethernet Interface" section on page 3-68.

After it is configured, the network connection takes place between client software on a workstation computer and a server process within the router. The type of client software you use depends on the server process to use. The Cisco IOS XR software supports the following client and server services:

- Telnet clients can connect to a Telnet server in the router. The Telnet server is disabled by default and can be enabled with the **telnet ipv4 server** or **telnet ipv6 server** command in global configuration mode.
- Secure Shell (SSH) clients can connect to an SSH server in the router. The SSH server is disabled by default and can be enabled with the **ssh server** command in global configuration mode. The SSH server handles both Secure Shell Version 1 (SSHv1) and SSHv2 incoming client connections for both IPv4 and IPv6 address families. The SSHv2 client is enhanced and can now execute commands remotely without invoking a secure interactive session.

To start a Telnet network connection, start the Telnet client software with a command similar to the following:

telnet *ManagementEthernetInterfaceIPaddress*

For specific instructions on connecting to the router through a Telnet or SSH client, see the instructions for that software.

Ask your system administrator for the IP address of the Management Ethernet interface.

When the Telnet session is established, the router prompts you to log in, as described in the "Logging In to a Router or an SDR" section on page 3-36.

Logging In to a Router or an SDR

The login process can require users to enter a password or a username and password before accessing the router CLI. The user groups to which your username is assigned determine which commands you can use.

If you log in to a router with a single SDR configured (this is the default configuration), you can manage the entire router.. If you log in to a named SDR, you can manage only that SDR. For more information on SDRs, see the "Secure Domain Routers" section on page 3-27.

When you log in, the username and password may be validated by any of the following services:

- Usernames configured on the router (**username** command in global configuration mode)
- Root-system usernames configured on the owner SDR
- Passwords configured for the router console and auxiliary ports (**password** or **secret** command in line configuration mode)
- RADIUS server
- TACACS+ server

The username and password validation method that your router uses is determined by the router configuration. For information on configuring username and password validation methods, see *Cisco IOS XR System Security Configuration Guide for the Cisco XR 12000 Series Router*. For information on which username and password to use, see your system administrator.

To log in to the router, enter your username and password when prompted. For example:

```
User Access Verification

Username: iosxr
Password: password
RP/0/0/CPU0:router#
```

Note Passwords are case sensitive. To log in to an SDR using a root-system username from the owner SDR, enter the username in the following format: *username*@admin. To support admin login, local database authentication must be enabled with the **aaa authentication login remote local** command. For more information, see *Cisco IOS XR System Security Configuration Guide for the Cisco XR 12000 Series Router*.

After you log in, the router displays the CLI prompt, which is described in the "CLI Prompt" section on page 3-37. The command set that you can use is determined by the privileges assigned to your username. For information on how privileges are assigned to usernames, see *Cisco IOS XR System Security Configuration Guide for the Cisco XR 12000 Series Router*.

CLI Prompt

After you log in, you see the CLI prompt for the Cisco IOS XR software. This prompt identifies the router or SDR to which you are issuing commands. The CLI prompt represents the path, through the router, to the CPU that executes the commands you enter. The syntax for the CLI prompt is: *type*/*rack*/*slot*/*module*: *router-name*#. Table 3-1 describes the CLI prompt.

Table 3-1 CLI Prompt Description

Prompt Syntax Component	Description
type	Type of interface or card with which you are communicating. For most user communication tasks, the type is "RP".
rack	Rack number. In a standalone router, the rack number is always "0".

Table 3-1 CLI Prompt Description (continued)

Prompt Syntax Component	Description
slot	Slot in which the RP or DRP is installed. In a Cisco XR 12000 Series Router, the physical slot number can be 0 to 15, and there can be multiple SDRs, each of which is represented by an RP.
module	Entity on a card that executes user commands or communicates with a port (interface). For executing commands from the EXEC prompt, the module is the "CPU0" of the RP. "CPU0" also controls the forwarding and operating system (OS) functions for the system. DRPs have two processors: CPU0 and CPU1.
router-name	Hostname of the router or SDR. The hostname is usually defined during initial configuration of the router, as described in the "Configuring the SDR Hostname" section on page 3-66.

For example, the following prompt indicates that the CLI commands are executed on the RP in rack 0, slot RP0, by the "CPU0" module on a router named "router":

```
RP/0/0/CPU0:router#
```

User Access Privileges

When you log in to the router, your username and password are used to determine if you are authorized to access the router. After you successfully log in, your username is used to determine which commands you are allowed to use. The following sections provide information on how the router determines which commands you can use:

User Groups, Task Groups, and Task IDs

The Cisco IOS XR software ensures security by combining tasks a user wants to perform (task IDs) into groups, defining which router configuration and management functions users can perform. This policy is enabled by the definition of:

- User groups—Collection of users that share similar authorization rights on a router.
- Task groups—Definition of collection of tasks identified by unique task IDs for each class of action.
- Task IDs—Definition of permission to perform particular tasks; pooled into a task group that is then assigned to users.

The commands you can perform are defined by the user groups to which you belong. Within the Cisco IOS XR software, the commands for a particular feature, like access control lists, are assigned to tasks. Each task is uniquely identified by a task ID. To use a particular command, your username must be associated with the appropriate task ID.

The association between a username and a task ID takes place through two intermediate entities, the user group and task group.

The user group is a logical container used to assign the same task IDs to multiple users. Instead of assigning task IDs to each user, you can assign them to the user group. Then, you can assign users to that user group. When a task is assigned to a user group, you can define the access rights for the commands associated with that task. These rights include “read”, “write”, “execute”, and “notify”.

The task group is also a logical container, but it is used to group tasks. Instead of assigning task IDs to each user group, you assign them to a task group. This allows you to quickly enable access to a specific set of tasks by assigning a task group to a user group.

To summarize the associations, usernames are assigned to user groups, which are then assigned to task groups. Users can be assigned to multiple user groups, and each user group can be assigned to one or more task groups. The commands that a user can execute are all those commands assigned to the tasks within the task groups that are associated with the user groups to which the user belongs.

Users are not assigned to groups by default and must be explicitly assigned by an administrator.

The following example shows how you can display all task IDs available on the system with the **show task supported** command.

```
RP/0/RP0/CPU0:router# show task supported

bgp
ospf
hsrp
isis
route-map
route-policy
static
vrrp
cef
lpts
iep
rib
multicast
mpls-te
mpls-ldp
mpls-static
ouni
fabric
bundle
network
transport
```

```
ppp
hdlc
 --More--
```

Note Only the root-system users, root-lr users, or users associated with the WRITE:AAA task ID can configure task groups. (The root-lr user has the highest level of privileges in an SDR. In previous releases, SDRs were called logical routers [LRs].)

Predefined User Groups

Cisco IOS XR software includes a set of predefined user groups that meets the needs of most organizations. Table 3-2 describes predefined user groups.

Table 3-2 Predefined User Group Descriptions

User Group	Privileges
root-system	Display and execute all commands for all SDRs in the system.
root-lr	Display and execute all commands within a single SDR.
sysadmin	Perform system administration tasks for the router, such as maintaining where the core dumps are stored or setting up the NTP[1] clock.
serviceadmin	Perform service administration tasks for the router, such as configuring firewall and sbc.
netadmin	Configure network protocols, such as BGP[2] and OSPF[3] (usually used by network administrators).
operator	Perform day-to-day monitoring activities, and have limited configuration rights.
cisco-support	Debug and troubleshoot features (usually, used by Cisco Technical Support personnel).

1. NTP stands for Network Time Protocol
2. BGP stands for Border Gateway Protocol
3. Open Shortest Path First

Although the predefined user groups are sufficient for the needs of most organizations, administrators can configure their own groups. For more information, see *Cisco IOS XR System Security Configuration Guide for the Cisco XR 12000 Series Router.*

Displaying the User Groups and Task IDs for Your User Account

To display the user groups and task IDs associated with your account, enter the **show user** command in EXEC mode. Table 3-3 summarizes the options available for this command.

Table 3-3 Options to Display Information About Your Account

Command	Description
show user	Displays your user name.
show user group	Displays the user groups assigned to your account.
show user tasks	Displays the task IDs assigned to your account.
show user all	Displays all user groups and task ID information for your account.
show aaa usergroup *group-name*	Displays the task IDs assigned to a user group.

Examples

The following examples show how to view user privileges:

- show user Command: Example, page 3-41
- show user tasks Command: Example, page 3-41
- show user group Command: Example, page 3-42
- show aaa usergroup Command: Example, page 3-42

show user Command: Example

To display your username, enter the **show user** command:

```
RP/0/0/CPU0:router# show user

username1
```

show user tasks Command: Example

To display the tasks assigned to your account and your rights to those tasks, enter the **show user tasks** command:

```
RP/0/0/CPU0:router# show user tasks
Mon May 31 02:52:13.335 DST
Task:                  aaa  : READ    WRITE    EXECUTE    DEBUG
Task:                  acl  : READ    WRITE    EXECUTE    DEBUG
Task:                admin  : READ    WRITE    EXECUTE    DEBUG
Task:                ancp   : READ    WRITE    EXECUTE    DEBUG
Task:                 atm   : READ    WRITE    EXECUTE    DEBUG
Task:       basic-services  : READ    WRITE    EXECUTE    DEBUG
Task:                bcdl   : READ    WRITE    EXECUTE    DEBUG
Task:                 bfd   : READ    WRITE    EXECUTE    DEBUG
Task:                 bgp   : READ    WRITE    EXECUTE    DEBUG
Task:                boot   : READ    WRITE    EXECUTE    DEBUG
Task:              bundle   : READ    WRITE    EXECUTE    DEBUG
Task:                 cdp   : READ    WRITE    EXECUTE    DEBUG
```

```
Task:                  cef  : READ    WRITE    EXECUTE    DEBUG
Task:                  cgn  : READ    WRITE    EXECUTE    DEBUG
Task:        cisco-support  : READ    WRITE    EXECUTE    DEBUG (reserved)
Task:          config-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:      config-services  : READ    WRITE    EXECUTE    DEBUG
Task:               crypto  : READ    WRITE    EXECUTE    DEBUG
Task:                 diag  : READ    WRITE    EXECUTE    DEBUG
Task:              drivers  : READ    WRITE    EXECUTE    DEBUG
Task:                 dwdm  : READ    WRITE    EXECUTE    DEBUG
Task:                  eem  : READ    WRITE    EXECUTE    DEBUG
Task:                eigrp  : READ    WRITE    EXECUTE    DEBUG
Task:    ethernet-services  : READ    WRITE    EXECUTE    DEBUG
```

show user group Command: Example

To display the user groups assigned to your user account, enter the **show user group** command:

```
RP/0/0/CPU0:router# show user group
Mon May 31 02:53:59.933 DST
root-system, cisco-support
```

show user all Command: Example

To display all user groups and task ID information for your account, enter the **show user all** command:

```
RP/0/0/CPU0:router# show user all
Mon May 31 02:54:51.446 DST
Username: cisco
Groups: root-system, cisco-support
Authenticated using method local
User cisco has the following Task ID(s):

Task:                  aaa  : READ    WRITE    EXECUTE    DEBUG
Task:                  acl  : READ    WRITE    EXECUTE    DEBUG
Task:                admin  : READ    WRITE    EXECUTE    DEBUG
Task:                 ancp  : READ    WRITE    EXECUTE    DEBUG
Task:                  atm  : READ    WRITE    EXECUTE    DEBUG
Task:       basic-services  : READ    WRITE    EXECUTE    DEBUG
Task:                 bcdl  : READ    WRITE    EXECUTE    DEBUG
Task:                  bfd  : READ    WRITE    EXECUTE    DEBUG
Task:                  bgp  : READ    WRITE    EXECUTE    DEBUG
Task:                 boot  : READ    WRITE    EXECUTE    DEBUG
Task:               bundle  : READ    WRITE    EXECUTE    DEBUG
Task:                  cdp  : READ    WRITE    EXECUTE    DEBUG
Task:                  cef  : READ    WRITE    EXECUTE    DEBUG
Task:                  cgn  : READ    WRITE    EXECUTE    DEBUG
Task:        cisco-support  : READ    WRITE    EXECUTE    DEBUG (reserved)
Task:          config-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:      config-services  : READ    WRITE    EXECUTE    DEBUG
Task:               crypto  : READ    WRITE    EXECUTE    DEBUG
Task:                 diag  : READ    WRITE    EXECUTE    DEBUG
```

show aaa usergroup Command: Example

To display the rights assigned to a user group, enter the **show aaa usergroup** *group-name* command:

```
RP/0/0/CPU0:router# show aaa usergroup root-system
Mon May 31 02:56:45.975 DST
User group 'root-system'
  Inherits from task group 'root-system'
```

```
User group 'root-system' has the following combined set
  of task IDs (including all inherited groups):
Task:                  aaa  : READ    WRITE    EXECUTE    DEBUG
Task:                  acl  : READ    WRITE    EXECUTE    DEBUG
Task:                admin  : READ    WRITE    EXECUTE    DEBUG
Task:                 ancp  : READ    WRITE    EXECUTE    DEBUG
Task:                  atm  : READ    WRITE    EXECUTE    DEBUG
Task:       basic-services  : READ    WRITE    EXECUTE    DEBUG
Task:                 bcdl  : READ    WRITE    EXECUTE    DEBUG
Task:                  bfd  : READ    WRITE    EXECUTE    DEBUG
Task:                  bgp  : READ    WRITE    EXECUTE    DEBUG
Task:                 boot  : READ    WRITE    EXECUTE    DEBUG
Task:               bundle  : READ    WRITE    EXECUTE    DEBUG
Task:                  cdp  : READ    WRITE    EXECUTE    DEBUG
Task:                  cef  : READ    WRITE    EXECUTE    DEBUG
Task:                  cgn  : READ    WRITE    EXECUTE    DEBUG
Task:          config-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:      config-services  : READ    WRITE    EXECUTE    DEBUG
Task:               crypto  : READ    WRITE    EXECUTE    DEBUG
Task:                 diag  : READ    WRITE    EXECUTE    DEBUG
```

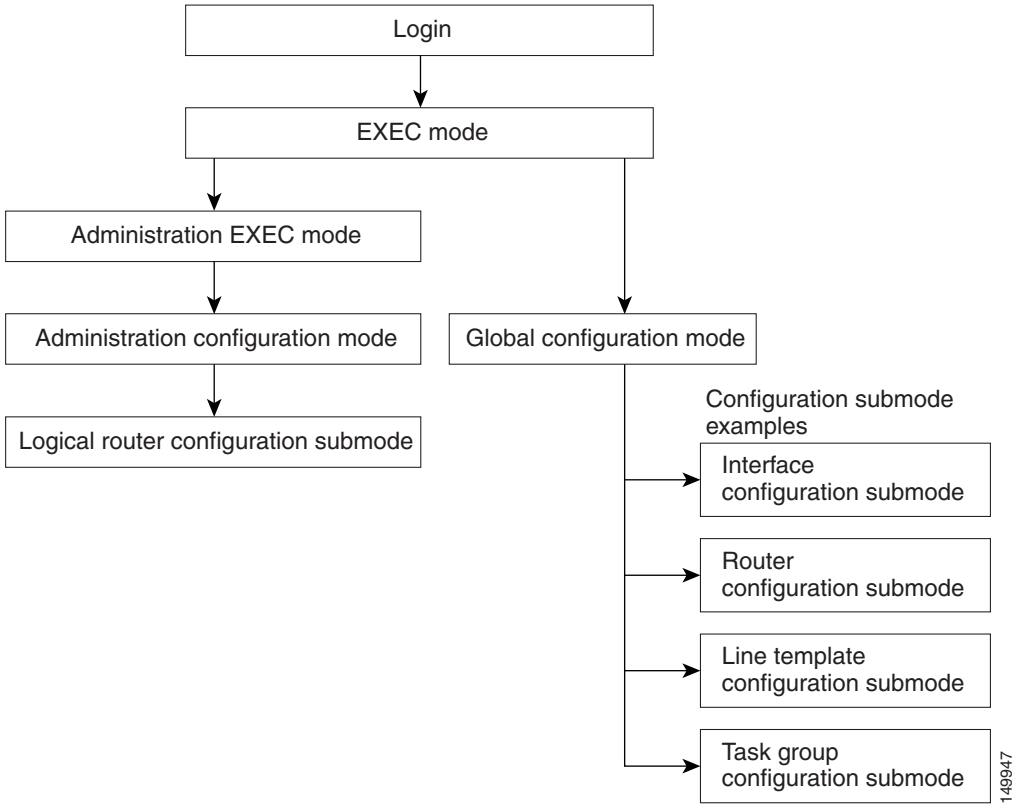
Navigating the Cisco IOS XR Command Modes

The Cisco IOS XR Software has different command modes. Each mode provides access to a subset of commands used to configure, monitor, and manage the router. Access to a mode is determined by your user group assignments. The following sections describe the navigation of the command modes:

- Identifying the Command Mode in the CLI Prompt, page 3-44
- Summary of Common Command Modes, page 3-45
- Entering EXEC Commands from a Configuration Mode, page 3-47
- Command Mode Navigation Example, page 3-48

Figure 3-4 illustrates the basic command mode navigation for the CLI. Only a small sample of the possible configuration modes is shown.

Figure 3-4 Example of Command Mode Navigation in Cisco IOS XR software



Identifying the Command Mode in the CLI Prompt

The command mode is identified in the CLI prompt after the router name.

When the router enters global configuration mode from the EXEC mode, the CLI prompt changes to include "(config)" after the router name:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)#
```

When the router enters interface configuration submode, the prompt changes to include "(config-if)" after the router name:

```
RP/0/0/CPU0:router(config)# interface POS 0/2/0/0
RP/0/0/CPU0:router(config-if)#
```

Summary of Common Command Modes

Table 3-4 summarizes the most common command modes of the Cisco IOS XR software and the associated CLI prompts.

Table 3-4 Common Command Modes and CLI prompts

Command Mode	Description
EXEC	Automatically places the router in EXEC mode when logging in to an SDR running the Cisco IOS XR software. Example: `RP/0/0/CPU0:router#` EXEC mode enables a basic set of commands to display the operational state of an SDR and the Cisco IOS XR software. Most CLI commands in EXEC mode do not change the SDR operation. The most common EXEC commands are **show** commands (to display SDR configuration or operational data) and **clear** commands (to clear or reset SDR counters). In EXEC mode, you can display the configuration of an SDRbut not the configuration of the system. The difference is that SDRs are defined in administration configuration mode, which is a submode of administration EXEC mode. SDRs are configured in global configuration mode. Additional commands are available depending on the access privileges (user groups) assigned to your username. Minimal privileges also include a small set of EXEC commands for connecting to remote devices, changing terminal line settings on a temporary basis, and performing basic tests.

Administration EXEC	Manages system resources. In administration EXEC mode, you can display the configuration of the system but not the configuration of an SDR. The difference is that SDRs are defined in administration configuration mode, which is a submode of administration EXEC mode. SDRs are configured in global configuration mode.
	Administration EXEC mode is used primarily to display system-wide parameters, configure the administration plane over the control Ethernet, and configure SDR. These operations are available only to users with the required root level access.
	From EXEC mode, use the **admin** command to enter administration EXEC mode:
	`RP/0/0/CPU0:router# admin`
	`RP/0/0/CPU0:router(admin)#`

Administration configuration	Allows you to create SDRs and assign system resources to SDRs. Multishelf systems are also configured in administration configuration mode.
	From administration EXEC mode, use the **configure** command to enter administration configuration submode:
	`RP/0/0/CPU0:router(admin)# configure`
	`RP/0/0/CPU0:router(admin-config)#`

Table 3-4 Common Command Modes and CLI prompts (continued)

Command Mode	Description

Global configuration	Global configuration mode is the starting point for SDR configuration. Commands entered in this mode affect the SDR as a whole, rather than just one protocol or interface. Global configuration mode is also used for entering configuration submodes to configure specific elements, such as interfaces or protocols.
	
	To enter global configuration mode, enter the **configure** command at the EXEC command prompt:
	
	`RP/0/0/CPU0:router# configure`
	`RP/0/0/CPU0:router(config)#`
	
	Note The system prompt changes to `router(config)` to indicate that the router is now in global configuration mode.

Configuration submodes	From the global configuration mode, you can also enter other, more specific command modes. These modes are available based on your assigned access privileges and include protocol-specific, platform-specific, and feature-specific configuration modes.
	
	In the following example, MPLS LDP configuration mode is entered from global configuration mode. The prompt for MPLS LDP configuration submode appears as config-ldp. The following command syntax is used for entering configuration MPLS LDP submode:
	
	`RP/0/0/CPU0:router# configure`
	`RP/0/0/CPU0:router(config)# mpls ldp`
	`RP/0/RP0/CPU0:router(config-ldp)#`
	
	Note The availability of any particular mode depends on the router features and the access rights of the individual user. For example, a configuration mode for configuring access servers is not available on most routers.

Interface configuration	The interface configuration submode is used to select and configure a hardware interface. To enter interface configuration mode from global configuration mode, use an **interface** command. An interface configuration command always follows an interface global configuration command, which defines the interface type. The following command syntax is used for entering interface configuration submode:
	
	`RP/0/0/CPU0:router# interface POS 0/2/0/0`
	`RP/0/RP0/CPU0:router(config-if)#`

Router configuration	The router configuration submode is used to select and configure a routing protocol, such as BGP, OSPF, or IS-IS. The **router** *protocol* [*protocol_options*] command syntax is used for entering router configuration submode.
	
	Replace *protocol* with the keyword for the protocol to configure. Replace *protocol_options* with any keywords and arguments required for that protocol. In the following example, the router enters the router configuration mode for BGP:
	
	`RP/0/0/CPU0:router# configure`
	`RP/0/0/CPU0:router(config)# router bgp 140`
	`RP/0/0/CPU0:router(config-bgp)#`

Table 3-4 Common Command Modes and CLI prompts (continued)

Command Mode	Description

Router submode configuration	Router configuration submodes are accessed from router configuration mode. The following command syntax is used for entering router address family configuration submode:
	`RP/0/0/CPU0:router(config)# router bgp 140`
	`RP/0/0/CPU0:router(config-bgp)# address-family ipv4 multicast`
	`RP/0/0/CPU0:router(config-bgp-af)#`
	
	For more information, see the following Cisco documents:
	• *Cisco IOS XR Routing Configuration Reference for the Cisco XR 12000 Series Router*
	• *Cisco IOS XR Routing Command Reference for the Cisco XR 12000 Series Router*
	•

ROM Monitor (ROMMON)	The ROM Monitor is a bootstrap program that initializes the hardware and boots the system when a router is powered on or reset. ROM Monitor mode is also known as *ROMMON*, which reflects the CLI prompt for the mode. `rommon B1>` During normal operation, users do not interact with ROMMON. This mode is accessed only by manually interrupting the boot process and placing the system in ROMMON. Once in ROMMON, you can perform ROM Monitor tasks, including reinstallation of the Cisco IOS XR software, password recovery, and other diagnostic tasks. The ROM Monitor CLI mode is accessible only from a terminal connected directly to the Console port of the primary RP, a terminal-modem connection to the AUX port, or through a terminal server. For information and instructions on using ROM Monitor mode, see Cisco IOS XR ROM Monitor Guide for the Cisco XR 12000 Series Router. For information and instructions on using ROM Monitor mode, see *Cisco IOS XR ROM Monitor Guide* for information and instructions on using ROM Monitor mode.

Entering EXEC Commands from a Configuration Mode

EXEC commands can be executed from any configuration mode by preceding the command with the **do** keyword. Executing EXEC commands from a configuration mode allows you to display the state of the system without exiting the configuration mode. For example:

```
RP/0/0/CPU0:router(config)# do show version


Mon May 31 02:52:07.240 DST

Cisco IOS XR Software, Version 4.1.0[Default]
Copyright (c) 2010 by Cisco Systems, Inc.

ROM: System Bootstrap, Version 12.0(20090226:235859) [rtauro-sw30346-33S 1.23dev(0.35)]
DEVELOPMENT
SOFTWARE
Copyright (c) 1994-2009 by cisco Systems,  Inc.

PE6_C12406 uptime is 2 weeks, 6 days, 10 hours, 4 minutes
System image file is "disk0:c12k-os-mbi-4.1.0/mbiprp-rp.vm"
```

```
cisco 12406/PRP (7457) processor with 2097152K bytes of memory.
7457 processor at 1266Mhz, Revision 1.2

1 Cisco 12000 Series Performance Route Processor
1 Cisco 12000 Series - Multi-Service Blade Controller
1 Cisco 12000 4-Port ISE ATM Over SONET OC3/STM-1 Controller (4 ATM)
1 Cisco 12000 Series SPA Interface Processor-601/501/401
1 Cisco 12000 Series SPA Interface Processor-600
3 Management Ethernet
6 PLIM_QOS
1 MgmtMultilink
5 SONET/SDH
2 T3
3 Multilink network interface(s)
28 T1
21 E1
21 Serial network interface(s)
5 GigabitEthernet/IEEE 802.3 interface(s)
4 Asynchronous Transfer Mode
1018k bytes of non-volatile configuration memory.
3623092k bytes of disk0: (Sector size 512 bytes).
3623092k bytes of disk1: (Sector size 512 bytes).
65536k bytes of Flash internal SIMM (Sector size 256k).

Configuration register on node 0/0/CPU0 is 0x102
Boot device on node 0/0/CPU0 is disk0:
Package active on node 0/0/CPU0:
c12k-fwding, V 4.1.0[DT_IMAGE], Cisco Systems, at disk0:c12k-fwding-4.1.0
    Built on Thu May  6 17:07:57 DST 2010
    By sjc-lds-364 in /auto/ioxbuild6/production/4.1.0.DT_IMAGE/c12k/workspace for pie
--More--
```

Command Mode Navigation Example

The following steps provide an example of command mode navigation:

Step 1 Start a session by logging in to the router and entering EXEC mode, as shown in the following example:

```
router con0_0_CPU0 is now available


Press Enter to get started.


User Access Verification

Username: iosxr
Password:<secret>
RP/0/0/CPU0:router#
```

From EXEC mode you can issue EXEC commands or enter global configuration mode. Examples of EXEC commands are the **show** commands used to display system status and **clear** commands to clear counters or interfaces.

Step 2 Add **?** at the end of the prompt, or after a command, to display the available options:

```
RP/0/0/CPU0:router# show ?

MgmtMultilink         Show trace data for the multilink controller component
  aaa                   Show AAA configuration and operational data
```

```
  access-lists          Access lists
  address-pool          Local address pool
  adjacency             Adjacency information
  af-ea                 AF-EA Platform details
  aliases               Display alias commands
  app-obj               APP-OBJ Show Commands
  aps                   SONET APS information
  aqsm                  AQSM show commands
  aqsmlib               AQSMLIB show commands
  arm                   IP ARM information
  arp                   ARP show commands
  arp-gmp               ARP show commands
  asic-errors           ASIC error information
  atc                   Attractor Cache related
  atm                   ATM information
  atm-vcm               Show atm_vcm component
  attractor             Show commands for attractor process
  attribute             IM Attributes operations information
  auto-rp               Auto-RP Commands
  bcdl                  Show Bulk Content DownLoader information
  bfd                   BFD information
 --More--
```

Note The commands available depend on the router mode and your user group assignments.

Step 3 If you belong to a user group that has configuration privileges, you can place the router in the global configuration mode by entering the **configure** command:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)#
```

Step 4 From global configuration mode, you can place the router in a configuration submode, such as interface configuration mode or a protocol-specific configuration mode.

In the following example, the router enters interface configuration mode and the user selects a POS interface for configuration. The command syntax is **interface** *type rack/slot/module/port*.

```
RP/0/0/CPU0:router(config)# interface POS 0/2/0/4
RP/0/0/CPU0:router(config-if)#
```

The command mode prompt changes from (config) to (config-if) and you can now enter configuration commands for the specified interface.

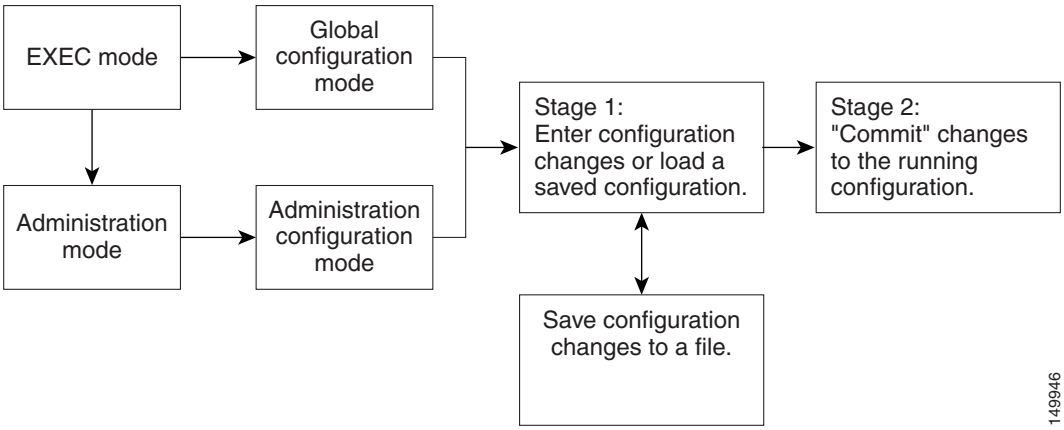
Step 5 To exit interface configuration mode and return to global configuration mode, enter the **exit** command. To return to EXEC mode, enter the **end** command.

Managing Configuration Sessions

In the Cisco IOS XR software, you cannot change the running (active) configuration directly. Enter configuration changes into an inactive target configuration. When the target configuration is ready for use, you can apply that configuration to the router with the **commit** command. This two-stage process allows you to make, edit, and verify configuration changes before impacting the actual running state of the router.

Figure 3-5 shows the two-stage configuration process.

Figure 3-5 Two-Stage Configuration Process



Global configuration mode is used to configure SDR features, such as routing protocols and interfaces. Administration configuration mode is used to assign hardware components to SDRs.

The following sections describe the management options for configuration sessions:

Displaying the Active Configuration Sessions

Before you start a configuration session, you should check if there are other configuration sessions in progress. More than one user can open a target configuration session at a time, allowing multiple users to work on separate target configurations.

The procedure for viewing the active configuration sessions depends on the type of configuration session. For administration configuration sessions, which assign hardware components in SDRs and multishelf systems, you must be in administration EXEC mode to view the active administration configuration sessions. For SDR configuration sessions, you must be in EXEC mode to view the active SDR configuration sessions.

To view the active administration configuration sessions, connect to the DSC and enter the **show configuration sessions** command in administration EXEC mode, as shown in the following example:

```
RP/0/0/CPU0:router# admin
RP/0/0/CPU0:router(admin)# show configuration sessions

Session                   Line        User      Date                      Lock
00000201-002180dd-00000000  vty0        cisco     Thu Mar 16 14:47:08 2006
```

To view the active SDR configuration sessions, connect to the appropriate SDR and enter the **show configuration sessions** command in EXEC mode, as shown in the following example:

```
RP/0/0/CPU0:router# show configuration sessions
```

```
Current Configuration Session                  Line       User     Date
Lock
00000201-002180dd-00000000  vty0        test      Thu Mar 16 13:16:17 2006
00000201-001b307a-00000000  vty2        cisco     Thu Mar 16 13:16:17 2006  *
```

If an asterisk (*) appears in the Lock column, the user is using an exclusive configuration session and you cannot start a configuration session until the exclusive configuration session closes. For more information, see the "Starting an Exclusive Configuration Session" section on page 3-53.

Note Configuration sessions for administration configuration and each SDR are managed independently. For example, if a user locks the administration configuration, you can still configure an SDR if other users have not locked a configuration session for that SDR.

Starting a Configuration Session

When you place the router in global configuration mode or administration configuration mode using the **configure** command, a new target configuration session is created. The target configuration allows you to enter, review, and verify configuration changes without impacting the running configuration.

Note The target configuration is not a copy of the running configuration. It has only the configuration commands entered during the target configuration session.

While in configuration mode, you can enter all Cisco IOS XR software commands supported in that configuration mode. Each command is added to the target configuration. You can view the target configuration by entering the **show configuration** command in configuration mode. The target configuration is not applied until you enter the **commit** command, as described in the "Committing Changes to the Running Configuration" section on page 3-62.

You can save target configurations to disk as nonactive configuration files. These saved files can be loaded, further modified, and committed at a later time. For more information, see the "Saving the Target Configuration to a File" section on page 3-60.

Examples

The following examples show how to manage configuration sessions:

Simple Owner SDR Configuration: Example

The following example shows a simple owner SDR configuration session in which the target configuration is created and previewed in global configuration mode:

```
RP/0/0/CPU0:router # configure
RP/0/0/CPU0:router(config)# interface POS 0/2/0/1
RP/0/0/CPU0:router(config-if)# description faq
RP/0/0/CPU0:router(config-if)# ipv4 address 10.10.10.10 255.0.0.0
RP/0/0/CPU0:router(config-if)# show configuration

Building configuration....
interface POS0/0/0/1
 description faq
```

```
ipv4 address 10.10.10.10 255.0.0.0
end
```

Simple Administration Configuration Session: Example

The following example shows a simple administration configuration session in which the target configuration is created and previewed in administration configuration mode:

```
RP/0/0/CPU0:router# admin
RP/0/0/CPU0:router(admin)# configure
RP/0/0/CPU0:router(admin-config)# sdr test
RP/0/0/CPU0:router(admin-config-sdr:test)# location 0/1/SP
RP/0/0/CPU0:router(admin-config-sdr:test)# show configuration

Building configuration...
sdr test
 location 0/1/SP
!
end
```

Starting an Exclusive Configuration Session

An exclusive configuration session allows you to configure the administration configuration or an SDR and lock out all users from committing configuration changes until you are done. Other users can still create and modify a target configuration, but they cannot commit those changes to the running configuration until you exit your exclusive configuration session.

During regular configuration sessions, the running configuration is locked whenever a commit operation is being performed. This automatic locking ensures that each commit operation is completed before the next one begins. Other users receive an error message if they attempt to commit a target configuration while another commit operation is under way.

To start an exclusive configuration session for an SDR, connect to that SDR and enter the **configure exclusive** command:

```
RP/0/0/CPU0:router# configure exclusive
RP/0/0/CPU0:router(config)#
```

Note If the configuration is already locked by another user, the **configure exclusive** command fails. To view locked and unlocked configuration sessions, see the "Displaying the Active Configuration Sessions" section on page 3-51.

To start an exclusive configuration session for the administration configuration, connect to the DSC and enter the **configure exclusive** command in administration EXEC mode:

```
RP/0/0/CPU0:router# admin
RP/0/0/CPU0:router(admin)# configure exclusive
RP/0/0/CPU0:router(admin-config)#
```

The running configuration is unlocked when the user who started the exclusive configuration session exits the configuration mode, as described in the "Ending a Configuration Session" section on page 3-65.

Displaying Configuration Details with show Commands

The following sections describe the following tasks:

Displaying the Running Configuration

The running configuration is the committed configuration that defines the router operations, and it is divided into the administration configuration and an SDR configuration for each SDR. The portion of the running configuration that you can view depends on the current CLI mode and SDR connection.

In EXEC mode and global configuration mode, you can view the SDR configuration for the SDR to which you are connected. When you are connected to the DSC and operating in administration EXEC and administration configuration mode, you can view the administration configuration, which includes hardware assignments for SDRs and multishelf systems.

To display the SDR portion of the running configuration, connect to the appropriate SDR and enter the **show running-config** command in EXEC or global configuration mode, as shown in the following example:

```
RP/0/0/CPU0:router(config)# show running-config

Building configuration...
!! Last configuration change at 11:05:38 UTC Mon May 02 2005 by cisco
!
hostname router
logging console debugging
telnet ipv4 server max-servers 5
username iosxr
 password 7 011F0706
 group root-system
 group cisco-support
!
ntp
 interface Loopback99
  broadcast
 !
 interface Loopback999
  broadcast
 !
 interface Loopback9999
  broadcast
 !
 authenticate
 max-associations 2000
!
interface Loopback0
 ipv4 address 10.1.2.3 255.255.0.0
 load-interval 0
!
```

```
interface Loopback1
 ipv4 address 10.4.5.6 255.255.0.0
!
interface Loopback7
 load-interval 0
!
interface Loopback2000
 load-interval 0
!
interface Loopback2001
 load-interval 0
!
interface Loopback2003
 load-interval 0
!
interface MgmtEth0/RP1/CPU0/0
 ipv4 address 10.11.12.13 255.255.0.0
!
interface POS0/0/0/0
 shutdown
!
interface POS0/0/0/1
 shutdown
!
interface POS0/0/0/2
 shutdown
!
interface POS0/0/0/3
 shutdown
!
interface POS0/3/0/0
 shutdown
!
interface POS0/3/0/1
 shutdown
!
interface POS0/3/0/2
 shutdown
!
interface POS0/3/0/3
 shutdown
!
interface preconfigure MgmtEth0/RP0/CPU0/0
 shutdown
!
router static
 address-family ipv4 unicast
 0.0.0.0/0 MgmtEth0/RP1/CPU0/0
 !
!
end
```

To display the administration portion of the running configuration, connect to the DSC and enter the **show running-config** command in administration EXEC or administration configuration mode, as shown in the following example:

```
RP/0/0/CPU0:router(admin)# show running-config

Building configuration...
sdr test
 location 0/1/* primary
!
username username1
 secret 5 $1$SegP$9jcoyk09S5cM.h/tX36yj.
 group root-system
!
end
```

Displaying a Sanitized Version of the Running Configuration

A sanitized running configuration report displays the contents of the running configuration without installation specific parameters. Some configuration details, such as IP addresses, are replaced with different addresses. The sanitized configuration can be used to share a configuration without exposing the configuration details.

In EXEC and global configuration mode, you can view the sanitized SDR configuration for the SDR to which you are connected. When you are connected to the SDR and operating in administration EXEC and administration configuration mode, you can view the sanitized administration configuration, which includes hardware assignments for SDRs.

To display the sanitized SDR portion of the running configuration, enter the **show running-config sanitized** command in EXEC or global configuration mode, as shown in the following example:

```
RP/0/0/CPU0:router(config)# show running-config sanitized

Building configuration...
!! Last configuration change at 11:05:38 UTC Mon May 02 2005 by <removed>
!
hostname <removed>
logging console debugging
telnet ipv4 server max-servers 5
username <removed>
 password 7 <removed>
 group root-system
 group cisco-support
!
ntp
 interface Loopback99
  broadcast
 !
 interface Loopback999
  broadcast
 !
 interface Loopback9999
  broadcast
 !
 authenticate
 max-associations 2000
!
interface Loopback0
 ipv4 address 10.0.0.0 255.0.0.0
 load-interval 0
!
```

```
interface Loopback1
 ipv4 address 10.0.0.0 255.0.0.0
!
interface Loopback7
 load-interval 0
!
interface Loopback2000
 load-interval 0
!
interface Loopback2001
 load-interval 0
!
interface Loopback2003
 load-interval 0
!
interface MgmtEth0/RP1/CPU0/0
 ipv4 address 10.0.0.0 255.0.0.0
!
interface POS0/0/0/0
 shutdown
!
interface POS0/0/0/1
 shutdown
!
interface POS0/0/0/2
 shutdown
!
interface POS0/0/0/3
 shutdown
!
interface POS0/3/0/0
 shutdown
!
interface POS0/3/0/1
 shutdown
!
interface POS0/3/0/2
 shutdown
!
interface POS0/3/0/3
 shutdown
!
interface preconfigure MgmtEth0/RP0/CPU0/0
 shutdown
!
router static
 address-family ipv4 unicast
 0.0.0.0/0 MgmtEth0/RP1/CPU0/0
 !
!
end
```

To display the sanitized administration portion of the running configuration, connect to the DSC and enter the **show running-config sanitized** command in administration EXEC or administration configuration mode, as shown in the following example:

```
RP/0/0/CPU0:router(admin)# show running-config sanitized
Mon May 31 21:35:14.902 DST
Building configuration...
!! IOS XR Admin Configuration 4.1.0
sdr <removed>
 location 0/1/*
 location 0/4/* primary
!
```

```
username <removed>
 group root-system
 group cisco-support
 secret 5 <removed>
!
end
```

Displaying the Target Configuration

The target configuration includes the configuration changes that have been entered but not yet committed. These changes are not yet part of the running configuration.

You can view the target configuration in global configuration and administration configuration modes. You cannot view the target configuration in EXEC modes because the target configuration must be committed or abandoned before returning to EXEC or administration EXEC mode.

To display the target configuration changes you have entered for an SDR, enter the **show configuration** command in global configuration mode or in any submode, as shown in the following example:

```
RP/0/0/CPU0:router(config-if)# show configuration

Building configuration...
interface POS0/3/0/3
 description faq
 ipv4 address 10.1.1.1 255.0.0.0
end
```

To display the target administration configuration changes you have entered, enter the **show configuration** command in administration configuration mode or in any submode, as shown in the following example:

```
RP/0/0/CPU0:router(admin-config-sdr:test)# show configuration

Building configuration...
sdr test
 location 0/1/* primary
!
end
```

Displaying a Combined Target and Running Configuration

Although the target and running configurations remain separate until the target configuration is committed, you can preview the combined target and running configuration without committing the changes. The combined configuration shows what the new running configuration will look like after the changes from the target configuration are committed. It does not represent the actual running configuration.

You can preview the combined configuration in global configuration and administration configuration modes. You cannot preview the combined configuration in EXEC modes because the target configuration must be committed or abandoned before returning to EXEC or administration EXEC mode.

To display the combined target and running configuration, enter the **show configuration merge** command in any configuration mode.

Note The **merge** option does not appear in command help until the target configuration contains at least one configuration change.

The following example shows how to display the active SDR configuration (**show running-config**), configure an interface, and display the merged configuration:

```
RP/0/0/CPU0:router# show running-config

Building configuration...
!! Last configuration change at 16:52:49 UTC Sun March 10 2004 by cisco
!
hostname router
shutdown
end

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface POS 0/3/0/3
RP/0/RP0/CPU0:router(config-if)# description faq
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.0.0.0

RP/0/RP0/CPU0:router(config)# show configuration merge
Building configuration...
!! Last configuration change at 16:52:49 UTC Sun March 10 2004 by cisco
!
hostname router
interface POS0/3/0/3
 description faq
 ipv4 address 10.1.1.1 255.0.0.0
 shutdown
end
```

Displaying Configuration Error Messages and Descriptions

Configuration changes are automatically verified during the commit operation, and a message appears if one or more configuration entry fails. To display an error message and description for a failed configuration, enter the **show configuration failed** command.

Note You can view configuration errors only during the current configuration session. If you exit configuration mode after the commit operation, the configuration error information is lost.

In the following example, an error is introduced in global configuration mode and the error information appears after the commit operation fails:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# taskgroup alr
RP/0/0/CPU0:router(config-tg)# description this is a test of an invalid taskgroup
RP/0/0/CPU0:router(config-tg)# commit

% Failed to commit one or more configuration items. Please use 'show configuration failed'
to view the errors

RP/0/0/CPU0:router(config-tg)# show configuration failed

!! CONFIGURATION FAILED DUE TO SEMANTIC ERRORS
taskgroup alr
!!% Usergroup/Taskgroup names cannot be taskid names
!
```

Displaying Configuration Error Messages Without Descriptions

Configuration changes are automatically verified during the commit operation, and a message appears if one or more configuration entry fails. To display only the error message (without a description) for a failed configuration, enter the **show configuration failed noerror** command, as shown in the following example:

```
RP/0/0/CPU0:router(config-tg)# show configuration failed noerror

!! CONFIGURATION FAILED DUE TO SEMANTIC ERRORS
taskgroup alr
!
```

Note You can view configuration errors only during the current configuration session. If you exit configuration mode after the commit operation, the configuration error information is lost.

Displaying Configuration Error Messages Produced While Loading a Configuration

To display any syntax errors found in a configuration loaded with the **load** command, enter the **show configuration failed load** command.

Saving the Target Configuration to a File

Target configurations can be saved to a separate file without committing them to the running configuration. Target configuration files can then be loaded at a later time and further modified or committed.

To save the configuration changes in the target configuration to a file, enter the **save configuration** *device***:** command. Replace the *device* argument with the name of the device on which you want to store the file (for example, disk0). After you enter this command, the router prompts you to enter a filename. If you enter only a filename, the file is stored in the root directory of the device. To store the file in a directory, enter the directory path and filename when prompted. We recommend that you specify the cfg file extension for easy identification. This suffix is not required, but it can help locate target configuration files, for example:

```
myconfig.cfg
```

The following example shows a target configuration file saved to the usr/cisco directory of disk0:

```
RP/0/0/CPU0:router(admin-config)# save configuration disk0:

Mon May 31 21:52:13.237 DST
Destination file name (control-c to abort): [/running-config]?/usr/cisco/test.cfg
Building configuration.
1 lines built in 1 second
[OK]
```

You can also save a configuration to a file using the **show configuration | file** *filename* command:

```
RP/0/0/CPU0:router(config)#show configuration | file abc.cfg
Thu Jul 22 23:03:04.722 DST
Building configuration...

[OK]
```

Loading the Target Configuration from a File

To populate the target configuration with the contents of a previously saved configuration file, go to global configuration or administration configuration mode and enter the **load** *filename* command. Consider the following when entering the *filename* argument:

- Specifies the configuration file to be loaded into the target configuration.
- If the full path of the file is not specified, the router attempts to load the file from the root directory on the device.

The following example shows a target configuration file loaded into the current configuration session. The current configuration session is populated with the contents of the file.

```
RP/0/0/CPU0:router(config)# load disk0:/usr/cisco/test.cfg

Loading.
77 bytes parsed in 1 sec (76)bytes/sec
```

Loading an Alternative Configuration at System Startup

When a router is reset or powered on, the last running configuration is loaded and used to operate the router.

You can load an alternative configuration during system boot. For information and instructions on this process, see *Cisco IOS XR ROM Monitor Guide for the Cisco XR 12000 Series Router.*

Clearing All Changes to a Target Configuration

To clear changes made to the target configuration without terminating the configuration session, enter the **clear** command in global configuration mode or administration configuration mode. This command deletes any configuration changes that have not been committed.

In the following example, the user configures an interface but does not commit it. After reviewing the changes to the target configuration with the **show configuration** command, the user decides to remove the changes and start over by entering the **clear** command:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface POS 0/3/0/1
RP/0/0/CPU0:router(config-if)# description this is my interface
RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.0.0.0
RP/0/0/CPU0:router(config-if)# shutdown
RP/0/0/CPU0:router(config-if)# exit

RP/0/0/CPU0:router(config)# show configuration

Building configuration...
interface POS0/3/0/1
 description this is my interface
 ipv4 address 10.1.1.1 255.0.0.0
 shutdown
end

RP/0/0/CPU0:router(config)# clear
RP/0/0/CPU0:router(config)# show configuration
Building configuration...
end
```

Committing Changes to the Running Configuration

The changes in the target configuration do not become part of the running configuration until you enter the **commit** command. When you commit a target configuration, you can use the **commit** command to do either of the following:

- Merge the target configuration with the running configuration to create a new running configuration.
- Replace the running configuration with the target configuration.

Note If you try to end a configuration session without saving your changes to the running configuration with the **commit** command, you are prompted to save the changes. For more information, see the "Ending a Configuration Session" section on page 3-65.

To commit target configuration changes to the running configuration, enter the **commit** command by itself or with one or more of the options described in Table 3-5.

Table 3-5 Commit Command Options

Command	Description
commit	(Default) Merges the target configuration with the running configuration and commits changes only if all changes in the target configuration pass the semantic verification process. If any semantic errors are found, none of the configuration changes takes effect.
commit best-effort	Merges the target configuration with the running configuration and commits only valid changes (best effort). Some configuration changes might fail due to semantic errors.
commit comment *line*	(Optional) Assigns a comment to a commit. • This text comment appears in the commit entry displayed with the **show configuration commit list** [**detail**] command. • The *line* argument is the text for the optional comment or label. • The **comment** option must appear at the end of the command line. If multiple options are entered, all text after the **comment** option is treated as a comment.
commit confirmed *seconds*	(Optional) Commits the configuration in global configuration mode on a trial basis for a minimum of 30 seconds and a maximum of 300 seconds (5 minutes). • During the trial configuration, enter **commit** to confirm the configuration. If you do not enter the **commit** command, the router reverts to the previous configuration when the trial time period expires. • The **confirmed** option is not available in administration configuration mode.
commit label *line*	(Optional) Assigns a meaningful label. This label appears in the output for the **show configuration commit list** [**detail**] command instead of the numeric label. • The *line* argument is the text for the optional comment or label.

Table 3-5 Commit Command Options (continued)

Command	Description
commit force	(Optional) Merges the target configuration with the running configuration and allows a configuration commit in low-memory conditions. A low-memory warning occurs when a user attempts to commit a target configuration that exceeds the default capacity of the router. The recommended resolution to such a warning is to remove configurations using the **no** commands. **Caution** The **force** option can cause the router to experience severe problems if low-memory conditions occur. The **force** option should be used only to remove configurations.
commit replace	(Optional) Replaces the contents of the running configuration with the target configuration.

Examples

The following examples illustrate how to commit a configuration:

- Committing a Configuration from Global Configuration Mode: Example, page 3-63
- Committing a Configuration from Administration Configuration Mode: Example, page 3-63

Committing a Configuration from Global Configuration Mode: Example

In the following example, the default **commit** command is entered in global configuration mode:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface POS 0/0/0/2
RP/0/0/CPU0:router(config-if)# description faq
RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.0.0.0
RP/0/0/CPU0:router(config-if)# commit

RP/0/0/0:Aug  6 09:26:17.781 : %LIBTARCFG-6-COMMIT Configuration committed by user
'cisco'.   Use 'show configuration commit changes 1000000124' to view the changes.
```

Note The preceding message is stored in the log and appears only if logging is configured to display on screen.

Committing a Configuration from Administration Configuration Mode: Example

In the following example, the **commit** command is entered with the **label** and **comment** keywords in administration configuration mode:

```
RP/0/0/CPU0:router# admin
RP/0/0/CPU0:router(admin)# configure
RP/0/0/CPU0:router(admin-config)# sdr test
RP/0/0/CPU0:router(admin-config-sdr:test)# location 0/1/* primary
RP/0/0/CPU0:router(admin-config-sdr:test)# commit label test comment This is a test
RP/0/0/CPU0:router(admin-config)# show configuration commit list detail

   1) CommitId: 2000000018                Label: test
```

```
      UserId:   user1                       Line:  vty1
      Client:   CLI                         Time:  23:45:40 UTC Tue Mar 07 2006
      Comment:   This is a test
.
.
.
```

Note Configuration files are stored on the same flash disk as the boot image. Access these configurations only through the CLI commands for configuration management, history, and rollback. Direct modification or deletion of these files can result in lost router configurations.

Reloading a Failed Configuration

If the router displays a configuration failure message when you attempt to commit a configuration change, the configuration changes are not lost. While you remain in global configuration mode or administration configuration mode, you can load the configuration changes into the target configuration, correct the errors, and commit the changes.

To load a failed configuration, go to global configuration or administration configuration mode and enter the **load configuration failed commit** command, as shown in the following example:

```
RP/0/0/CPU0:router(config)# load configuration failed commit
RP/0/0/CPU0:router(config)# show configuration

Building configuration...
taskgroup alr
!
end
```

In the preceding example, the **show configuration** command displays the target configuration, which includes the failed configuration.

Note The failed configuration is discarded if you exit global configuration mode or administration configuration mode without recovering the configuration. After recovery, correct and commit the configuration or save it to a file to avoid losing it.

Exiting a Configuration Submode

When you have finished configuration changes in a configuration submode, such as the interface or SDR configuration submodes, you can return to the previous configuration mode and continue making configuration changes. To exit a configuration submode, enter the **exit** command, as shown in the following example:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface POS 0/3/0/1
RP/0/0/CPU0:router(config-if)# description this is my interface
RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.0.0.0
RP/0/0/CPU0:router(config-if)# exit
RP/0/0/CPU0:router(config)#
```

Note If you use the **exit** command to exit global configuration or administration configuration mode, the router prompts you to save changes, discard changes, or cancel the action, as described in the next section.

Returning Directly to Configuration Mode from a Submode

When you have finished configuration changes in a configuration submode, such as the interface or SDR configuration submodes, you can skip all intermediate submodes and return to the top-level configuration mode and continue making configuration changes. To return to configuration mode, enter the **root** command, as shown in the following example:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# router static
RP/0/0/CPU0:router(config-static)# address-family ipv4 unicast
RP/0/0/CPU0:router(config-static-afi)# root
RP/0/0/CPU0:router(config)#
```

Ending a Configuration Session

You can use any of the following methods to end a configuration session:

- Enter the **exit** command in global configuration or administration configuration mode
- Enter the **end** command in any configuration mode or submode
- Press **Ctrl-Z**

Note If you enter the **exit** command in a configuration submode, the command returns you to the parent configuration level.

If you end a configuration session without committing the configuration changes, the router prompts you to save changes, discard changes, or cancel the action, as shown in the following example:

```
RP/0/0/CPU0:router(config-if)# end

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:
```

Respond to the prompt by entering one of the following options:

- **yes**—Commit the configuration changes and exit configuration mode
- **no**—Exit configuration mode without committing the configuration changes
- **cancel**—Remain in configuration mode without committing the configuration changes

Note In EXEC mode, the **exit** command logs the user out of the system.

Aborting a Configuration Session

When you abort a configuration session, any changes are discarded and the configuration session ends. No warning is given before the configuration changes are deleted.

The **abort** command in global configuration mode, discards configuration changes and returns to EXEC mode. In administration configuration mode, the **abort** command discards configuration changes and returns to administration EXEC mode. To abort a configuration session, enter the **abort** command, as shown in the following example:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# hostname host1
RP/0/0/CPU0:router(config)# interface POS 0/2/0/2
```

```
RP/0/0/CPU0:router(config-if)# description this is my interface
RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.0.0.0
RP/0/0/CPU0:router(config-if)# shutdown
RP/0/0/CPU0:router(config-if)# abort
RP/0/0/CPU0:router#
```

Configuring the SDR Hostname

The hostname identifies an SDR on the network. Although devices can be uniquely identified by their Layer 2 and Layer 3 addresses (such as an IP address), it is often simpler to remember network devices by an alphanumeric “hostname.” This name is used in the CLI prompt and default configuration filenames and to identify the SDR on the network.

To configure the hostname, enter the **hostname** command with the SDR name as shown in the following example:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# hostname SDR_SJ
RP/0/0/CPU0:router(config)# commit

RP/0/0/CPU0:Apr  7 00:07:33.246 : config[65669]: %LIBTARCFG-6-COMMIT : Configuration
committed by user 'user_a'.   Use 'show configuration commit changes 1000000067' to view
the changes.
RP/0/0/CPU0:SDR_SJ(config)#
```

The preceding example sets the SDR name to SDR_SJ.

Note No blanks or spaces are permitted as part of a name. Do not expect case to be preserved. Uppercase and lowercase characters look the same to many Internet software applications. It may seem appropriate to capitalize a name the same way you might if you were writing, but conventions dictate that computer names appear all lowercase. For more information, see RFC 1178, *Choosing a Name for Your Computer*.

Configuring the Management Ethernet Interface

The Management Ethernet interface on the RPs is used to connect the router to a network for remote management using a Telnet client, the Simple Network Management Protocol (SNMP), or other management agents. The following sections provide information on the Management Ethernet interface:

- Specifying the Management Ethernet Interface Name in CLI Commands, page 3-67
- Displaying the Available Management Ethernet Interfaces, page 3-67
- Configuring the Management Ethernet Interface, page 3-68

Specifying the Management Ethernet Interface Name in CLI Commands

Before you can configure the Management Ethernet interface, you must know the Management Ethernet interface name, which is defined using the following syntax: *typerack/slot/module/port*. Table 3-6 describes the Management Ethernet interface name syntax.

Table 3-6 Management Ethernet Interface Name Syntax Description

Syntax Component	Description
type	Interface type for a Management Ethernet port is "MgmtEth."
rack	Chassis number of the rack. In a single-shelf system, the *rack* is always "0".
slot	Physical slot of the RP or DRP on which the interface is located. For a Cisco XR 12000 Series Router, the PRPs may be installed in slots 0 through 15, depending on the router model.
module	On an RP, the module is "CPU0". DRPs have two processors, so the *module* is either "CPU0" and "CPU1".
port	On Cisco XR 12000 Series Routers, there are three Ethernet ports on PRP-2 cards. The Ethernet ports are labeled ETH 0, ETH 1, and ETH 2. For the ETH 0 port, specify **0**; for the ETH 1 port, specify **1**; and for the ETH 2 port, specify **2**. Specify 0 for the MGMT ETH interface on an RP or DRP.

Table 3-7 shows examples of Management Ethernet interface names for a single-shelf system.

Table 3-7 Management Ethernet Interface Names for Single-Shelf Systems

Management Interface	Interface Name	Example
Cisco XR 12000 Series Router PRP in slot 0, port ETH0	MgmtEth0/0/CPU0/0	router(config)# **interface MgmtEth0/0/CPU0/0**
Cisco XR 12000 Series Router PRP in slot 0, port ETH1	MgmtEth0/0/CPU0/1	router(config)# **interface MgmtEth0/0/CPU0/1**
Cisco XR 12000 Series Router PRP in slot 1, port ETH0	MgmtEth0/1/CPU0/0	router(config)# **interface MgmtEth0/1/CPU0/0**
Cisco XR 12000 Series Router PRP in slot 1, port ETH1	MgmtEth0/1/CPU0/1	router(config)# **interface MgmtEth0/1/CPU0/1**

Displaying the Available Management Ethernet Interfaces

To display the router interfaces, enter the **show interfaces brief** command in EXEC mode:

```
RP/0/0/CPU0:router# show interfaces brief
Mon May 31 22:03:33.039 DST

                 Intf       Intf        LineP              Encap  MTU        BW
```

```
                Name       State       State               Type (byte)    (Kbps)
--------------------------------------------------------------------------------
                 Lo0          up          up           Loopback  1500        0
                 Nu0          up          up               Null  1500        0
              ti1019          up          up        TUNNEL_GRE  1500      100
             ti10100          up          up        TUNNEL_GRE  1500      100
             ti10200          up          up        TUNNEL_GRE  1500      100
               tt100        down        down             TUNNEL  1500        0
              tt1060          up          up             TUNNEL  1500        0
           PO0/6/0/0          up          up               HDLC  4474   155520
           PO0/6/0/1          up          up               HDLC  4474   155520
           PO0/6/0/2  admin-down  admin-down               HDLC  4474   155520
           PO0/6/0/3          up          up               HDLC  4474   155520
           Te0/6/1/0          up          up               ARPA  1514 10000000
           PO0/6/4/0  admin-down  admin-down               HDLC  4474   622080
           PO0/6/4/1  admin-down  admin-down               HDLC  4474   622080
           PO0/6/4/2  admin-down  admin-down               HDLC  4474   622080
           PO0/6/4/3  admin-down  admin-down               HDLC  4474   622080
           PO0/6/4/4          up          up               HDLC  4474   622080
           PO0/6/4/5          up          up               HDLC  4474   622080
           PO0/6/4/6          up          up               HDLC  4474   622080
           PO0/6/4/7  admin-down  admin-down               HDLC  4474   622080
           Gi0/6/5/0  admin-down  admin-down               ARPA  1514  1000000
           Gi0/6/5/1          up          up               ARPA  2014  1000000
           Gi0/6/5/2          up          up               ARPA  2014  1000000
           Gi0/6/5/3  admin-down  admin-down               ARPA  1514  1000000
           Gi0/6/5/4          up          up               ARPA  2014  1000000
           Gi0/6/5/5          up          up               ARPA  2014  1000000
           Gi0/6/5/6          up          up               ARPA  2014  1000000
           Gi0/6/5/7          up          up               ARPA  2014  1000000
     Mg0/RP0/CPU0/0          up          up               ARPA  1514   100000
     Mg0/RP1/CPU0/0          up          up               ARPA  1514   100000
```

Configuring the Management Ethernet Interface

To use the Management Ethernet interface for system management and remote communication, you must configure an IP address and a subnet mask for the interface. To have the interface communicate with devices on other networks (such as remote management stations or TFTP servers), you need to configure a default route for the router.

Tip For information on additional configuration options for the Management Ethernet interface, see *Cisco IOS XR Interface and Hardware Component Configuration Guide for the Cisco XR 12000 Series Router.*

Prerequisites

To configure the Ethernet Management port for network communications, you must enter the interface network addresses and subnet mask. Consult your network administrator or system planner for this information.

SUMMARY STEPS

1. **configure**
2. **interface MgmtEth** *rack*/*slot*/**CPU0**/*port*

3. **ipv4 address** *ipv4-address subnet-mask*
4. **no shutdown**
5. **exit**
6. **router static address-family ipv4 unicast 0.0.0.0/0** *default-gateway*
7. **commit**
8. **end**
9. **show interfaces MgmtEth***rack*/*slot*/**CPU0/***port*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**interface MgmtEth***rack***/***slot***/CPU0/***port***** **Example:** `RP/0/0/CPU0:router(config)# interface MgmtEth0/RP0/CPU0/0`	Enters interface configuration mode and specifies the Management Ethernet interface of the primary RP. • The syntax is **interface** *typerack/slot/module/port*: Table 3-6 describes the command parameters.
Step 3	**ipv4 address** *ipv4-address subnet-mask* **Example:** `RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.0.0.0`	Assigns an IP address and subnet mask to the interface.
Step 4	**no shutdown** **Example:** `RP/0/0/CPU0:router(config-if)# no shutdown`	Places the interface in an "up" state.
Step 5	**exit**	Exits the Management Ethernet interface configuration mode.
Step 6	**router static address-family ipv4 unicast 0.0.0.0/0** *default-gateway* **Example:** `RP/0/0/CPU0:router (config)# router static address-family ipv4 unicast 0.0.0.0/0 12.25.0.1`	Configures a default route to use for communications with devices on other networks. • Replace *default-gateway* with the IP address of the local gateway that can be used to reach other networks. • This default route applies to all interfaces. You might need to configure additional static routes to support your network. For more information on configuring static routes, see *Cisco IOS XR Routing Configuration Guide for the Cisco XR 12000 Series Router.*
Step 7	**commit** **Example:** `RP/0/0/CPU0:(config)# commit`	Commits the target configuration to the running configuration.

	Command or Action	Purpose
Step 8	**end** **Example:** `RP/0/0/CPU0:router(config)# 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]:` • 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.
Step 9	**show interfaces MgmtEth***rack***/***slot***/CPU0/***port*** **Example:** `RP/0/0/CPU0:router# show interfaces MgmtEth0/RP0/CPU0/0`	Displays interface details to verify the settings.

Examples

The following example shows how the Management Ethernet interface on the RP in slot RP1 is configured with an IP address.

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface MgmtEth0/0/CPU0/0
RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.255.255.0
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# commit
RP/0/0/CPU0:router(config-if)# end
RP/0/0/CPU0:router#
RP/0/0/CPU0:router# show interfaces MgmtEth 0/0/CPU0/0

MgmtEth0/0/CPU0/0 is up, line protocol is up
Interface state transitions: 1
  Hardware is Management Ethernet, address is 0011.93ef.e8e6 (bia 0011.93ef.e8e6)
  Description: Connected to Lab LAN
  Internet address is 172.29.52.70/24
  MTU 1514 bytes, BW 100000 Kbit
     reliability 255/255, txload 0/255, rxload 0/255
  Encapsulation ARPA,
  Half-duplex, 100Mb/s, 1000BASE-T, link type is autonegotiation
  output flow control is off, input flow control is off
  loopback not set,
  ARP type ARPA, ARP timeout 04:00:00
  Last input 00:00:00, output 00:00:00
  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
     31371 packets input, 1922996 bytes, 153 total input drops
     0 drops for unrecognized upper-level protocol
     Received 19457 broadcast packets, 0 multicast packets
              12 runts, 0 giants, 0 throttles, 0 parity
     61 input errors, 27 CRC, 12 frame, 0 overrun, 0 ignored, 0 abort
     12869 packets output, 878236 bytes, 0 total output drops
```

```
Output 5 broadcast packets, 0 multicast packets
0 output errors, 0 underruns, 0 applique, 0 resets
0 output buffer failures, 0 output buffers swapped out
1 carrier transitions
```

Related Documents

Related Topic	Document Title
Additional information about configuring management interfaces	*Advanced Configuration and Modification of the Management Ethernet Interface on Cisco IOS XR Software* module of *Cisco IOS XR Interface and Hardware Component Configuration Guide for the Cisco XR 12000 Series Router*

Manually Setting the Router Clock

Generally, if the system is synchronized by a valid outside timing mechanism, such as a Network Time Protocol (NTP) or VINES clock source, you do not need to set the software clock. Use the **clock set** command for initial configuration or if a network time source is not available.

The **clock timezone** command should be entered before the clock is set because it defines the difference between the system time and Coordinated Universal Time (UTC). When you set the time, you set the system time, and the router uses the **clock timezone** command setting to translate that time to UTC. The system internally keeps time in UTC. When you enter the **show clock** command, the router displays the system time.

To manually set the router clock, follow these steps:

SUMMARY STEPS

1. **configure**
2. **clock timezone** *zone hours-offset*
3. **commit**
4. **end**
5. **clock set** *hh*:*mm*:*ss dd mm yyyy*
6. **clock update-calendar**
7. **show clock**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**clock timezone** *zone hours-offset* **Example:** `RP/0/0/CPU0:router(config)# clock timezone pst -8`	Sets the time zone for the router clock. • **clock timezone** command should be entered before the clock is set because it defines the difference between the system time and UTC. **Note** The system time is the time that appears when you enter the **show clock** command. • *zone*—Name of the time zone to be displayed when standard time is in effect. • *hours-offset*—Difference in hours from UTC.
Step 3	**commit** **Example:** `RP/0/0/CPU0:router(config-if)# commit`	Commits the target configuration to the running configuration.
Step 4	**end** **Example:** `RP/0/0/CPU0:router(config-if)# end`	Ends the configuration session and returns to EXEC mode.
Step 5	**clock set** *hh:mm:ss dd mm yyyy* **Example:** `RP/0/0/CPU0:router# clock set 14:12:00 10 dec 2008`	Sets the system software clock.
Step 6	**clock update-calendar** **Example:** `RP/0/0/CPU0:router# clock update-calendar`	Updates the hardware clock (calendar clock) with the new clock settings. • It is battery operated and runs continuously, even if the router is powered off or rebooted.
Step 7	**show clock** **Example:** `RP/0/0/CPU0:router# show clock`	Displays the clock setting. • Use this command to verify the settings.

Examples

The following example shows how the manual system clock is configured:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# clock timezone pst -8
RP/0/0/CPU0:router(config)# commit
RP/0/0/CPU0:router(config)# end
```

```
RP/0/0/CPU0:router# clock set 14:12:00 10 dec 2008
14:12:00.090 PST Wed Dec 02 2008
RP/0/0/CPU0:router# clock update-calendar
RP/0/0/CPU0:router# show clock
14:12:00.090 PST Wed Dec 02 2008
```

Related Documents

Related Topic	Document Title
Descriptions of the clock commands	*Clock Commands on Cisco IOS XR Software* module of *Cisco IOS XR System Management Command Reference for the Cisco XR 12000 Series Router*
Commands used to configure NTP	*NTP Commands on Cisco IOS XR Software* module of *Cisco IOS XR System Management Command Reference for the Cisco XR 12000 Series Router*
Configuration of NTP	*Implementing NTP on Cisco IOS XR Software* module of *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router*

Where to Go Next

When you have completed the configuration procedures in this chapter, consider the following resources for additional configuration documentation:

- For information on configuring additional general router features, see Configuring Additional Router Features
- For information on using the Cisco IOS XR software more efficiently, see CLI Tips, Techniques, and Shortcuts
- For information on configuring interfaces, see the hardware documents listed in the "Related Documents" section on page x.

CHAPTER 4

Configuring Additional Router Features

This chapter contains instructions and information for entering basic configurations using the command-line interface (CLI).

Contents

Configuring the Domain Name and Domain Name Server

Configure a domain name and Domain Name Server (DNS) for your router to contact other devices on your network efficiently. Use the following guidelines:

- To define a default domain name that the Cisco IOS XR software uses to complete unqualified hostnames (names without a dotted-decimal domain name), use the **domain-name** command in global configuration mode.
- To specify the address of one or more name servers to use for name and address resolution, use the **domain name-server** command in global configuration mode. If no name server address is specified, the default name server is 255.255.255.255 so the DNS lookup can be broadcast to the local network segment. If a DNS server is in the local network, it replies. If not, there might be a server that knows how to forward the DNS request to the correct DNS server.
- Use the **show hosts** command in EXEC mode to display the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses.

To configure the DNS and DNS server, follow these steps:

SUMMARY STEPS

1. **configure**
2. **domain name** *domain-name-of-organization*
3. **domain name-server** *ipv4-address*

4. **end** or **commit**
5. **show hosts**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**domain name** *domain-name-of-organization* **Example:** `RP/0/0/CPU0:router(config)# domain name cisco.com`	Defines a default domain name used to complete unqualified hostnames.
Step 3	**domain name-server** *ipv4-address* **Example:** `RP/0/0/CPU0:router(config)# domain name-server 192.168.1.111`	Specifies the address of a name server to use for name and address resolution (hosts that supply name information). **Note** You can enter up to six addresses, but only one for each command.

Step 4	**end**
	or
	commit
	
	Example:
	`RP/0/0/CPU0:router(config)# end`
	or
	`RP/0/0/CPU0:router(config)# 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 hosts** **Example:** `RP/0/0/CPU0:router(config)# show hosts`	Displays all configured name servers.

Examples

The following example shows how the domain name and DNS are configured:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# domain name cisco.com
RP/0/0/CPU0:router(config)# domain name-server 10.1.1.1
RP/0/0/CPU0:router(config)# commit
RP/0/0/CPU0:router(config)# end
RP/0/0/CPU0:router# show hosts

Default domain is cisco.com
Name/address lookup uses domain service
Name servers: 10.1.1.1
```

Related Documents

Related Topic	Document Title
Complete descriptions of the domain services commands	*Implementing Host Services and Applications on Cisco IOS XR Software* module in *Cisco IOS XR IP Addresses and Services Configuration Guide for the Cisco XR 12000 Series Router*

Configuring Telnet, HTTP, and XML Host Services

For security reasons, some host services are disabled by default. You can enable Host services, such as Telnet, XML, and HTTP by using the commands described in this section. Host services provide the following features:

- Enabling the Telnet server allows users to log in to the router using IPv4 or IPv6 Telnet clients.
- Enabling the XML agent enables XML Common Object Request Broker Architecture (CORBA) agent services so that you can manage and configure the router using an XML interface.

Prerequisites

Ensure the following prerequisites are met before configuring Telnet, HTTP, and XML host services:

- For the XML and HTTP host services, the Manageability package must be installed and activated on the router.
- To enable the Secure Socket Layer (SSL) of the HTTP and XML services, the security package must be installed and activated on the router.

See *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router* for information on installing and activating packages.

Note This process enables the Telnet, HTTP, and XML host services on the Management Ethernet interfaces. For more information on how to enable these services on other inband interfaces, see *Implementing Management Plane Protection in Cisco IOS XR Software* module in *Cisco IOS XR System Security Configuration Guide for the Cisco XR 12000 Series Router.*

SUMMARY STEPS

1. **configure**
2. **interface MgmtEth** *interface-path-id*
   ipv4 address *ipv4-address subnetmask*
3. **ipv4 virtual address** *ipv4-address subnetmask*
4. **end** or **commit**
5. **exit**
6. **configure**
7. **telnet** {**ipv4** | **ipv6**} **server max-servers** *limit*
8. **http server**
9. **xml agent**
10. **end** or **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/0/CPU0:router# configure`	Enters global configuration mode.

Step 2	**interface MgmtEth** *interface-path-id*
	ipv4 address *ipv4-address subnetmask*
	
	Example:
	Active RP
	`RP/0/0/CPU0:router(config)# interface MgmtEth0/RP0/CPU0/0`
	`RP/0/0/CPU0:router(config-if)# ipv4 address 172.29.52.75 255.255.255.0`
	`RP/0/RP0/CPU0:router(config-if)# no shut`
	`RP/0/RP0/CPU0:router(config-if)# exit`
	
	Standby RP
	`RP/0/0/CPU0:router(config)# interface MgmtEth0/RP1/CPU0/0`
	`RP/0/0/CPU0:router(config-if)# ipv4 address 172.29.52.76 255.255.255.0`
	`RP/0/0/CPU0:router(config-if)# no shut`
	`RP/0/0/CPU0:router(config-if)# exit`
		Configures the Management Ethernet ports on the active and standby RPs.

Step 3	**ipv4 virtual address** *ipv4-address subnetmask* `RP/0/0/CPU0:router(config)# ipv4 virtual address 172.29.52.77 255.255.255.0`	Defines an IPv4 virtual address for the Management Ethernet interface.

	Command or Action	Purpose

Step 4	**end**
	or
	commit
	
	Example:
	`RP/0/0/CPU0:router(config)# end`
	or
	`RP/0/0/CPU0:router(config)# 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	**exit** **Example:** `RP/0/0/CPU0:router(config)# exit`	Exits global configuration mode.
Step 6	**configure** **Example:** `RP/0/0/CPU0:router# configure`	Enters global configuration mode.
Step 7	**telnet ipv4 server max-servers** *limit* or **telnet ipv6 server max-servers** *limit* **Example:** `RP/0/0/CPU0:router(config)# telnet ipv4 server max-servers 5`	Enables Telnet services on the router and specifies the maximum number of allowable Telnet servers.
Step 8	**http server** **Example:** `RP/0/0/CPU0:router(config)# http server`	Enables HTTP server on the router.

	Command or Action	Purpose

Step 9	**xml agent**
	
	Example:
	`RP/0/0/CPU0:router(config)# xml agent`
	`RP/0/0/CPU0:router(config)# xml agent tty`
		Enables XML requests on the router.

Step 10	**end**
	or
	commit
	
	Example:
	`RP/0/0/CPU0:router(config)# end`
	or
	`RP/0/0/CPU0:router(config)# 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.

Examples

The following example shows how the host services are enabled:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface MgmtEth0/RP0/CPU0/0
RP/0/0/CPU0:router(config)# ipv4 address 172.29.52.75 255.255.255.0
RP/0/0/CPU0:router(config)# ipv4 virtual address 172.29.52.77 255.255.255.0
RP/0/0/CPU0:router(config)# commit
RP/0/0/CPU0:router(config)# exit
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# telnet ipv4 server max-servers 5
RP/0/0/CPU0:router(config)# http server
RP/0/0/CPU0:router(config)# xml agent
RP/0/0/CPU0:router(config)# commit
```

Related Documents

Related Topic	Document Title
Installation and activation of the Manageability and Security Packages	*Upgrading and Managing Cisco IOS XR Software* module of *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router*
Descriptions of the HTTP and XML server commands	*Manageability Commands on Cisco IOS XR Software* module of *Cisco IOS XR System Management Command Reference for the Cisco XR 12000 Series Router*
Descriptions of the Telnet commands	*Host Services and Applications Commands on Cisco IOS XR Software* module of *Cisco IOS XR IP Addresses and Services Command Reference for the Cisco XR 12000 Series Router*

Managing Configuration History and Rollback

After each commit operation, the system saves a record of the committed configuration changes. This record contains only the changes made during the configuration session; it does not contain the complete configuration. Each record is assigned a unique ID, known as a *commit ID*.

When multiple commit IDs are present, you can use a commit ID to identify a previous configuration to which to return, or you can use the commit ID to load the configuration changes made during that configuration session. You can also load configuration changes from multiple commit IDs, and you can clear commit IDs. If you are thinking about rolling back the configuration to a specific commit ID, consider the following guidelines:

- You cannot roll back to a configuration that was removed because of package incompatibility. Configuration rollbacks can succeed only when the configuration passes all compatibility checks with the currently active Cisco IOS XR Software release.
- If the system finds an incompatible configuration during rollback, the operation fails and an error appears.

The Cisco IOS XR software automatically saves up to 100 of the most recent commit IDs. The following sections describe how to manage configuration changes and roll back to a previously committed configuration:

- Displaying the Commit IDs, page 4-82
- Displaying the Configuration Changes Recorded in a Commit ID, page 4-82
- Previewing Rollback Configuration Changes, page 4-83
- Rolling Back the Configuration to a Specific Rollback Point, page 4-83
- Rolling Back the Configuration over a Specified Number of Commits, page 4-84
- Loading Commit ID Configuration Changes to the Target Configuration, page 4-84
- Loading Rollback Configuration Changes to the Target Configuration, page 4-85
- Deleting Commit IDs, page 4-86

Displaying the Commit IDs

To display a history of up to 100 of the most recent commit IDs, enter the **show configuration commit list** command in EXEC or administration EXEC mode. Up to 100 of the most recent commit IDs are saved by the system. Each commit ID entry shows the user who committed configuration changes, the connection used to execute the commit, and commit ID time stamp.

The commit IDs are shown in the "Label/ID" column. The following example shows the **show configuration commit list** command display in EXEC and administration EXEC modes:

```
RP/0/0/CPU0:router# show configuration commit list

SNo. Label/ID    User      Line        Client      Time Stamp
~~~~ ~~~~~~~~    ~~~~      ~~~~        ~~~~~~      ~~~~~~~~~~
1    1000000219  cisco     vty0        CLI         12:27:50 UTC Wed Mar 22 2006
2    1000000218  cisco     vty1        CLI         11:43:31 UTC Mon Mar 20 2006
3    1000000217  cisco     con0_RP0_C  CLI         17:44:29 UTC Wed Mar 15 2006

RP/0/0/CPU0:router# admin
RP/0/0/CPU0:router(admin)# show configuration commit list

SNo. Label/ID    User      Line        Client      Time Stamp
~~~~ ~~~~~~~~    ~~~~      ~~~~        ~~~~~~      ~~~~~~~~~~
1    2000000022  cisco     vty1        CLI         15:03:59 UTC Fri Mar 17 2006
2    2000000021  cisco     con0_RP0_C  CLI         17:42:55 UTC Wed Mar 15 2006
3    2000000020  SYSTEM    con0_RP0_C  Setup Dial  17:07:39 UTC Wed Mar 15 2006
```

Displaying the Configuration Changes Recorded in a Commit ID

To display the configuration changes made during a specific commit session (commit ID), go to EXEC or administration EXEC mode and enter the **show configuration commit changes** command followed by a commit ID number. The easiest way to determine the commit ID is to enter the **show configuration commit changes ?** command first. In the following example, the command help is used to display the available commit IDs, and then the changes for a specific commit ID are displayed:

```
RP/0/0/CPU0:router(admin)# show configuration commit changes ?

  last        Changes made in the most recent <n> commits
  since       Changes made since (and including) a specific commit
  2000000020  Commit ID
  2000000021  Commit ID
  2000000022  Commit ID

RP/0/0/CPU0:router(admin)# show configuration commit changes 2000000020

Building configuration...
username cisco
 secret 5 $1$MgUH$xzUEW6jLfyAYLKJE.3p440
 group root-system
!
end
```

Previewing Rollback Configuration Changes

The **show configuration rollback changes** command allows you to preview the configuration changes that take place if you roll back the configuration to a specific commit ID. For example, if you want to roll back the configuration to a specific point, all configuration changes made after that point must be undone. This rollback process is often accomplished by executing the **no** version of commands that must be undone.

To display the prospective rollback configuration changes from the current configuration to a specific commit ID, go to EXEC or administration EXEC mode and enter the **show configuration rollback changes to** *commit ID* command. In the following example, the command help displays the available commit IDs, and then the rollback changes are displayed:

```
RP/0/0/CPU0:router# show configuration rollback changes to ?

  1000000217  Commit ID
  1000000218  Commit ID
  1000000219  Commit ID

RP/0/0/CPU0:router# show configuration rollback changes to 1000000218

Building configuration...
no interface Loopback100
interface POS0/1/0/0
 no ipv6 nd dad attempts
!
!
no route-policy xx
end
```

To display the prospective rollback configuration changes from the current configuration to a specified number of previous sessions, go to EXEC or administration EXEC mode and enter the **show configuration rollback changes last** *commit-range* command:

```
RP/0/0/CPU0:router# show configuration rollback changes last 2

Building configuration...
interface Loopback3
no description
no ipv4 address 10.0.1.1 255.0.0.0
exit
interface Loopback4
no description
no ipv4 address 10.0.0.1 255.0.0.0
end
```

In the preceding example, the command display shows the proposed rollback configuration changes for the last two commit IDs.

Rolling Back the Configuration to a Specific Rollback Point

When you roll back the configuration to a specific rollback point, you undo all configuration changes made during the session identified by the commit ID for that rollback point, and you undo all configuration changes made after that point. The rollback process rolls back the configuration and commits the rolled-back configuration. The rollback process also creates a new rollback point so that you can roll back the configuration to the previous configuration.

Tip To preview the commands that undo the configuration during a rollback, use the **show configuration rollback changes** command.

To roll back the router configuration to a previously committed configuration, go to EXEC or administration EXEC mode and enter the **rollback configuration to** *commit ID* command:

```
RP/0/0/CPU0:router# rollback configuration to 1000000220
Loading Rollback Changes.
Loaded Rollback Changes in 1 sec
Committing.
2 items committed in 1 sec (1)items/sec
Updating.
Updated Commit database in 1 sec
Configuration successfully rolled back to '1000000220'.
```

Rolling Back the Configuration over a Specified Number of Commits

When you roll back the configuration over a specific number of commits, you do not have to enter a specific commit ID. Instead, you specify a number *x*, and the software undoes all configuration changes made in the last *x* committed configuration sessions. The rollback process rolls back the configuration, commits the rolled-back configuration, and creates a new commit ID for the previous configuration.

Tip To preview the commands that undo the configuration during a rollback, use the **show configuration rollback changes** command.

To roll back to the last *x* commits made, go to EXEC or administration EXEC mode and enter the **rollback configuration last** *x* command; *x* is a number ranging from 1 to the number of saved commits in the commit database.

In the following example, a request is made to roll back the configuration changes made during the previous two commits:

```
RP/0/0/CPU0:router# rollback configuration last 2

Loading Rollback Changes.
Loaded Rollback Changes in 1 sec
Committing.
1 items committed in 1 sec (0)items/sec
Updating.
Updated Commit database in 1 sec
Configuration successfully rolled back 2 commits.
```

Loading Commit ID Configuration Changes to the Target Configuration

If the changes saved for a specific commit ID are close to what you want, but a rollback is not appropriate, you can load the configuration changes for a commit ID into the target configuration, modify the target configuration, and then commit the new configuration. Unlike the rollback process, the loaded changes are not applied until you commit them.

Note Unlike the rollback process, loading the commit ID configuration changes loads only the changes made during that commit operation. The load process does not load all changes made between the commit ID and the current committed configuration.

To load commit ID changes in the target configuration, go to global configuration or administration configuration mode and enter the **load commit changes** command with the commit ID number. In the following example, **show** commands are used to display the changes for a commit ID, the commit ID configuration is loaded into the target configuration, and the target configuration is displayed:

```
RP/0/0/CPU0:router# show configuration commit changes ?

  last        Changes made in the most recent <n> commits
  since       Changes made since (and including) a specific commit
  1000000217  Commit ID
  1000000218  Commit ID
  1000000219  Commit ID
  1000000220  Commit ID
  1000000221  Commit ID

RP/0/0/CPU0:router# show configuration commit changes 1000000219
Building configuration...
interface Loopback100
!
interface POS0/1/0/0
ipv6 nd dad attempts 50
!
end

RP/0/0/CPU0:router# config

RP/0/0/CPU0:router(config)# load commit changes 1000000219
Building configuration...
Loading.
77 bytes parsed in 1 sec (76)bytes/sec

RP/0/0/CPU0:router(config)# show configuration

Building configuration...
interface Loopback100
!
interface POS0/1/0/0
ipv6 nd dad attempts 50
!
end
```

Loading Rollback Configuration Changes to the Target Configuration

If the changes for a specific rollback point are close to what you want, but a rollback is not appropriate, you can load the rollback configuration changes into the target configuration, modify the target configuration, and then commit the new configuration. Unlike the rollback process, the loaded changes are not applied until you commit them.

Tip To display the rollback changes, enter the **show configuration rollback changes** command.

To load rollback configuration changes from the current configuration to a specific session, go to global configuration or administration configuration mode and enter the **load rollback changes to** *commit ID* command:

```
RP/0/0/CPU0:router(config)# load rollback changes to 1000000068

Building configuration...
Loading.
233 bytes parsed in 1 sec (231)bytes/sec
```

To load rollback configuration changes from the current configuration to a specified number of previous sessions, go to global configuration or administration configuration mode and enter the **load rollback changes last** *commit-range* command:

```
RP/0/0/CPU0:router(config)# load rollback changes last 6

Building configuration...
Loading.
221 bytes parsed in 1 sec (220)bytes/sec
```

In the preceding example, the command loads the rollback configuration changes for the last six commit IDs.

To load the rollback configuration for a specific commit ID, go to global configuration or administration configuration mode and enter the **load rollback changes** *commit ID* command:

```
RP/0/0/CPU0:router(config)# load rollback changes 1000000060

Building configuration...
Loading.
199 bytes parsed in 1 sec (198)bytes/sec
```

Deleting Commit IDs

You can delete the oldest configuration commit IDs by entering the **clear configuration commits** command in EXEC or administration EXEC mode. The **clear configuration commits** command must be followed by either the amount of disk space to reclaim or number of commit IDs to delete. To reclaim disk space from the oldest commit IDs, enter the **clear configuration commits** command followed by the **diskspace** keyword and number of kilobytes to reclaim:

```
RP/0/0/CPU0:router# clear configuration commits diskspace 50

Deleting 4 rollback points '1000000001' to '1000000004'
64 KB of disk space will be freed. Continue with deletion?[confirm]
```

To delete a specific number of the oldest commit IDs, enter the **clear configuration commits** command followed by the **oldest** keyword and number of commit IDs to delete:

```
RP/0/0/CPU0:router# clear configuration commits oldest 5

Deleting 5 rollback points '1000000005' to '1000000009'
80 KB of disk space will be freed. Continue with deletion?[confirm]
```

Configuring Logging and Logging Correlation

System messages generated by the Cisco IOS XR software can be logged to a variety of locations based on the severity level of the messages. For example, you could direct information messages to the system console and also log debugging messages to a network server.

In addition, you can define correlation rules that group and summarize related events, generate complex queries for the list of logged events, and retrieve logging events through an XML interface.

The following sections describe logging and the basic commands used to log messages in Cisco IOS XR software:

Logging Locations and Severity Levels

Table 4-1 shows error messages that can be logged to a variety of locations.

Table 4-1 Logging Locations for System Error Messages

Logging Destination	Command (Global Configuration Mode)
console	**logging console**
vty terminal	**logging monitor**
external syslog server	**logging trap**
internal buffer	**logging buffered**

Table 4-2 shows how you can log messages based on the severity level of the messages.

Table 4-2 Logging Severity Levels for System Error Messages

Level	Description
Level 0—Emergencies	System has become unusable.
Level 1—Alerts	Immediate action needed to restore system stability.
Level 2—Critical	Critical conditions that may require attention.
Level 3—Errors	Error conditions that may help track problems.
Level 4—Warnings	Warning conditions that are not severe.
Level 5—Notifications	Normal but significant conditions that bear notification.
Level 6—Informational	Informational messages that do not require action.
Level 7—Debugging	Debugging messages are for system troubleshooting only.

Alarm Logging Correlation

Alarm logging correlation is used to group and filter similar messages to reduce the amount of redundant logs and isolate the root causes of the messages.

For example, the original message describing the online insertion and removal (OIR) and system state being up or down can be reported, and all subsequent messages reiterating the same event can be correlated. When you create correlation rules, a common root event that is generating larger volumes of

follow-on error messages can be isolated and sent to the correlation buffer. An operator can extract all correlated messages for display later, should the need arise. For more information, see *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router.*

Configuring Basic Message Logging

Numerous options for logging system messages in Cisco IOS XR software are available. This section provides a basic example.

To configure basic message logging, follow these steps:

SUMMARY STEPS

1. **configure**
2. **logging** {*ip-address* | *hostname*}
3. **logging trap** *severity*
4. **logging console** [*severity*]
5. **logging buffered** [*severity* | *buffer-size*]
6. **commit**
7. **end**
8. **show logging**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**logging** {*ip-address* | *hostname*} **Example:** `RP/0/0/CPU0:router(config)# logging 10.1.1.1`	Specifies a syslog server host to use for system logging.
Step 3	**logging trap** *severity* **Example:** `RP/0/0/CPU0:router(config)# logging trap debugging`	Limits the logging of messages sent to syslog servers to only those messages at the specified level. • Table 4-2 shows a summary of the logging severity levels.
Step 4	**logging console** [*severity*] **Example:** `RP/0/0/CPU0:router(config)# logging console emergencies`	Logs messages on the console. • When a severity level is specified, only messages at that severity level are logged on the console. • Table 4-2 shows a summary of the logging severity levels.

	Command or Action	Purpose
Step 5	**logging buffered** [*severity* | *buffer-size*] **Example:** `RP/0/0/CPU0:router(config)# logging buffered 1000000`	Copies logging messages to an internal buffer. • Newer messages overwrite older messages after the buffer is filled. • Specifying a severity level causes messages at that level and numerically lower levels to be logged in an internal buffer. See Table 4-2 for a summary of the logging severity levels. • The buffer size is from 4096 to 4,294,967,295 bytes. Messages above the set limit are logged to the console.
Step 6	**commit** **Example:** `RP/0/0/CPU0:router(config)# commit`	Commits the target configuration to the router running configuration.
Step 7	**end** **Example:** `RP/0/0/CPU0:router(config)# end`	Ends the configuration session and returns to EXEC mode.
Step 8	**show logging** **Example:** `RP/0/0/CPU0:router# show logging`	Displays the messages that are logged in the buffer.

Examples

The following example shows how the basic message logging is configured.

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# logging 10.1.1.1
RP/0/0/CPU0:router(config)# logging trap debugging
RP/0/0/CPU0:router(config)# logging console emergencies
RP/0/0/CPU0:router(config)# logging buffered 1000000
RP/0/0/CPU0:router(config)# commit
RP/0/0/CPU0:router(config)# end
RP/0/0/CPU0:router# show logging

Syslog logging: enabled (162 messages dropped, 0 flushes, 0 overruns)
    Console logging: level emergencies, 593 messages logged
    Monitor logging: level debugging, 0 messages logged
    Trap logging: level debugging, 2 messages logged
    Logging to 10.1.1.1, 2 message lines logged
    Buffer logging: level debugging, 722 messages logged

Log Buffer (1000000 bytes):

RP/0/0/CPU0:Apr  8 19:18:58.679 : instdir[203]: %INSTALL-INSTMGR-6-INSTALL_OP
RP/0/0/CPU0:Apr  8 19:19:01.287 : instdir[203]: %INSTALL-INSTMGR-6-INSTALL_OP
RP/0/0/CPU0:Apr  8 19:22:15.658 : instdir[203]: %INSTALL-INSTMGR-6-INSTALL_OP
LC/0/1/CPU0:Apr  8 19:22:30.122 : sysmgr[74]: %OS-SYSMGR-7-INSTALL_NOTIFICATION
LC/0/6/CPU0:Apr  8 19:22:30.160 : sysmgr[74]: %OS-SYSMGR-7-INSTALL_NOTIFICATION
RP/0/0/CPU0:Apr  8 19:22:30.745 : sysmgr[79]: %OS-SYSMGR-7-INSTALL_NOTIFICATI
RP/0/0/CPU0:Apr  8 19:22:32.596 : sysmgr[79]: %OS-SYSMGR-7-INSTALL_NOTIFICATI
```

```
LC/0/1/CPU0:Apr  8 19:22:35.181 : sysmgr[74]: %OS-SYSMGR-7-INSTALL_FINISHED : s
LC/0/6/CPU0:Apr  8 19:22:35.223 : sysmgr[74]: %OS-SYSMGR-7-INSTALL_FINISHED : s
RP/0/0/CPU0:Apr  8 19:22:36.122 : sysmgr[79]: %OS-SYSMGR-7-INSTALL_FINISHED :
RP/0/0/CPU0:Apr  8 19:22:37.790 : sysmgr[79]: %OS-SYSMGR-7-INSTALL_FINISHED :
RP/0/0/CPU0:Apr  8 19:22:41.015 : schema_server[332]: %MGBL-SCHEMA-6-VERSIONC
RP/0/0/CPU0:Apr  8 19:22:59.844 : instdir[203]: %INSTALL-INSTMGR-4-ACTIVE_SOF
RP/0/0/CPU0:Apr  8 19:22:59.851 : instdir[203]: %INSTALL-INSTMGR-6-INSTALL_OP
 --More--
```

Disabling Console Logging

To disable console logging, enter the **logging console disable** command in global configuration mode.

Related Documents

Related Topic	Document Title
Configuration of system logging	*Implementing Logging Services on Cisco IOS XR Software* module of *Cisco IOS XR System Monitoring Configuration Guide for the Cisco XR 12000 Series Router*
Commands used to configure logging	*Logging Services Commands on Cisco IOS XR Software* module of *Cisco IOS XR System Monitoring Command Reference for the Cisco XR 12000 Series Router*
Configuration of alarm correlation and generating complex queries	*Implementing and Monitoring Alarms and Alarm Log Correlation on Cisco IOS XR Software* module of *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router*
Commands used to configure alarm correlation	*Alarm Management and Logging Correlation Commands on Cisco IOS XR Software* module of *Cisco IOS XR System Management Command Reference for the Cisco XR 12000 Series Router*
Retrieve logging events through an XML interface	*Cisco IOS XR XML API Guide for the Cisco XR 12000 Series Router*

Creating and Modifying User Accounts and User Groups

In the Cisco IOS XR software, users are assigned individual usernames and passwords. Each username is assigned to one or more user group, each of which defines display and configuration commands the user is authorized to execute. This authorization is enabled by default in the Cisco IOS XR software, and each user must log in to the system using a unique username and password.

The following section describe the basic commands used to configure users and user groups:

Displaying Details About User Accounts, User Groups, and Task IDs, page 4-91

For a summary of user accounts, user groups, and task IDs, see the "User Groups, Task Groups, and Task IDs" section on page 3-39.

Note The management of user accounts, user groups, and task IDs is part of the authentication, authorization, and accounting (AAA) feature. AAA is a suite of security features included in the Cisco IOS XR software. For more information on the AAA concepts and configuration tasks, see *Cisco IOS XR System Security Configuration Guide for the Cisco XR 12000 Series Router* and *Cisco IOS XR*

System Security Command Reference for the Cisco XR 12000 Series Router. For instructions to activate software packages, see *Cisco IOS XR System Management Configuration Guide for the Cisco XR 12000 Series Router*.

Displaying Details About User Accounts, User Groups, and Task IDs

Table 4-3 summarizes the EXEC mode commands used to display details about user accounts, user groups, and task IDs.

Table 4-3 Commands to Display Details About Users and User Groups

Command	Description
show aaa userdb *username*	Displays the task IDs and privileges assigned to a specific username. To display all users on the system, type the command without a username.
show aaa usergroup *usergroup-name*	Displays the task IDs and privileges that belong to a user group. To display all groups on the system, type the command without a group name.
show task supported	Displays all task IDs for the system. Only the root-system users, root-lr users, or users associated with the WRITE:AAA task ID can configure task groups.

CHAPTER 5

CLI Tips, Techniques, and Shortcuts

This chapter describes techniques for using the command-line interface (CLI) of the Cisco IOS XR software.

Contents

Note Commands can be entered in uppercase, lowercase, or mixed case. Only passwords are case sensitive. However, the Cisco Systems documentation convention presents commands in lowercase.

CLI Tips and Shortcuts

The following sections describe tips and shortcuts useful when using the CLI:

Entering Abbreviated Commands

You can abbreviate commands and keywords to the number of characters that allow a unique abbreviation. For example, the **configure** command can be abbreviated as **config** because the abbreviated form of the command is unique. The router accepts and executes the abbreviated command.

Using the Question Mark (?) to Display On-Screen Command Help

Use the question mark (**?**) to learn what commands are available and the correct syntax for a command. Table 5-1 summarizes the options for on-screen help.

Tip The space (or no space) before the question mark (**?**) is significant. If you include a space before the question mark, the system displays all available options for a command or CLI mode. If you do not include a space, the system displays a list of commands that begin with a particular character string.

Table 5-1 On-Screen Help Commands

Command	Description

partial-command **?**	Enter a question mark (**?**) at the end of a partial command to list the commands that begin with those characters.
	`RP/0/0/CPU0:router# co?`
	
	`configure copy`
	
	Note Do not include a space between the command and question mark.

?	Lists all commands available for a particular command mode.

command **?**	Include a space before the question mark (**?**) to list the keywords and arguments that belong to a command.
	`RP/0/0/CPU0:router# configure ?`
	
	`exclusive Configure exclusively from this terminal`
	`terminal Configure from the terminal`
	`<cr>`
	
	Note For most commands, the <cr> symbol indicates that you can execute the command with the syntax already entered. For the preceding example, press **Return** to enter global configuration mode.

command keyword **?**	Enter a question mark (?) after the keyword to list the next available syntax option for the command.
	`RP/0/0/CPU0:router# show aaa ?`
	
	`ikegroup Show local IKE group(s)`
	`locald locald sub system`
	`login login sub system`
	`task Show task information`
	`taskgroup Show all the local taskgroups configured in the system`
	`trace Show trace data for AAA sub system`
	`userdb Show all local users with the usergroups each belong to`
	`usergroup Show all the local usergroups configured in the system`
	
	Note Include a space between the keyword and question mark.

The following example shows how to add an entry to access list 99. The added entry denies access to all hosts on subnet 172.0.0.0 and ignores bits for IPv4 addresses that start within the range of 0 to 255. The following steps provide an example of on-screen command help:

Step 1 Enter the **access-list** command, followed by a space and a question mark, to list the available options for the command:

```
RP/0/0/CPU0:router(config)# ipv4 access-list ?

  log-update  Control access lists log updates
  ssm-acl     Access list name - maximum 32 characters
  bidir-acl   Access list name - maximum 32 characters
  WORD        Access list name - maximum 32 characters
```

Note The number ranges (within the angle brackets) are inclusive ranges.

Step 2 Enter the access list name **list1**, followed by a space and another question mark, to display the arguments that apply to the keyword and brief explanations:

```
RP/0/0/CPU0:router(config)# ipv4 access-list list1 ?

  log-update  Control access lists log updates
  ssm-acl     Access list name - maximum 32 characters
  bidir-acl   Access list name - maximum 32 characters
  WORD        Access list name - maximum 32 characters
RP/0/RP0/CPU0:router(config)#ipv4 access-list list1 ?
  <1-2147483646>  Sequence number for this entry
  deny            Specifies packets to reject
  permit          Specifies packets to forward
  remark          Comment for access list
  <cr>
RP/0/0/CPU0:router(config)#ipv4 access-list list1
```

Step 3 Enter the **deny option** and a question mark to see more command options:

```
RP/0/0/CPU0:router(config)#ipv4 access-list list1 deny ?

  <0-255>         An IPv4 Protocol Number
  A.B.C.D         Source IP address or prefix
  A.B.C.D/prefix  Source IP address and care bits
  ahp             Authentication Header Protocol
  any             Any source host
  eigrp           Cisco's EIGRP Routing Protocol
  esp             Encapsulation Security Payload
  gre             Cisco's GRE Tunneling
  host            A single source host
  icmp            Internet Control Message Protocol
  igmp            Internet Gateway Message Protocol
  igrp            Cisco's IGRP Routing Protocol
  ipinip          IP in IP tunneling
  ipv4            Any IPv4 Protocol
  nos             KA9Q NOS Compatible IP over IP Tunneling
  ospf            OSPF Routing Protocol
  pcp             Payload Compression Protocol
  pim             Protocol Independent Multicast
  sctp            Stream Control Transmission Protocol
  tcp             Transport Control Protocol
  udp             User Datagram Protocol
RP/0/0/CPU0:router(config)#ipv4 access-list list1 deny
```

Generally, uppercase letters represent variables (arguments).

Step 4 Enter an IP address, followed by a space and a question mark (?), to list additional options:

```
RP/0/0/CPU0:router(config)# ipv4 access-list list1 deny 172.31.134.0 ?

   A.B.C.D    Wildcard bits
  log        Log matches against this entry
  log-input  Log matches against this entry, including input interface
  <cr>

RP/0/0/CPU0:router(config)# ipv4 access-list list1 deny 172.31.134.0
```

The <cr> symbol by itself indicates that there are no more keywords or arguments.

Step 5 Press **Enter** to execute the command:

```
RP/0/0/CPU0:router(config)# ipv4 access-list list1 deny 172.31.134.0
```

Note The configuration does not become active until you enter the **commit** command to add the target configuration to the running configuration.

Completing a Partial Command with the Tab Key

If you do not remember a complete command name or want to reduce the amount of typing you have to perform, enter the first few letters of the command, then press the **Tab** key. If only one command begins with that character string, the system automatically completes the command for you. If the characters you entered indicate more than one command, the system beeps to indicate that the text string is not unique and the system provides a list of commands that match the text entered.

In the following example, the CLI recognizes **conf** as a unique string in EXEC mode and completes the command when you press the **Tab** key:

```
RP/0/0/CPU0:router# conf<Tab>
RP/0/0/CPU0:router# configure
```

The CLI displays the full command name. You must then press **Return** to execute the command. This feature allows you to modify or reject the suggested command.

In the next example, the CLI recognizes two commands that match the text entered:

```
RP/0/0/CPU0:router# co<Tab>
configure  copy
RP/0/0/CPU0:router# con<Tab>
RP/0/0/CPU0:router# configure
```

Tip If your keyboard does not have a Tab key, press **Ctrl-I** instead.

Identifying Command Syntax Errors

If an incorrect command is entered, an error message is returned with the caret (^) at the point of the error. In the following example, the caret appears where the character was typed incorrectly in the command:

```
RP/0/0/CPU0:router# configure termiMal
                                 ^
% Invalid input detected at '^' marker.
```

Note The percent sign (%) indicates the line in which the error message occurred.

To display the correct command syntax, enter the **?** after the command:

```
RP/0/0/CPU0:router# configure ?

  exclusive  Configure exclusively from this terminal
  terminal   Configure from the terminal
  <cr>
```

Using the no Form of a Command

Almost every configuration command has a **no** form. Depending on the command, the **no** form enables or disables a feature. For example, when configuring an interface, the **no shutdown** command brings up the interface, and the **shutdown** command shuts down the interface. The **username** command creates a new user, and the **no username** command deletes a user when entered with a valid username.

The Cisco IOS XR software command reference publications provide the complete syntax for the configuration commands and describe what the **no** form of a command does. For more information, see the "Related Documents" section on page x.

Editing Command Lines that Wrap

The CLI provides a wraparound feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. The first ten characters of the line are not shown, but it is possible to scroll back and check the syntax at the beginning of the command. To scroll back, press **Ctrl-B** or the **Left Arrow** key repeatedly, or press **Ctrl-A** to return directly to the beginning of the line.

In the following example, the **ipv4 access-list** command entry is too long to display on one line. When the cursor reaches the end of the line, the line is shifted to the left and redisplayed. The dollar sign ($) after the command prompt indicates that the line has been scrolled to the left and the beginning of the command is hidden.

```
RP/0/0/CPU0:router(config)# $s-list 101 permit tcp 172.31.134.5 255.255.255.0 172.31.135.0
```

In the next example, Ctrl-A is used to display the beginning of the command line, and the dollar sign at the end of the command line shows the command has been scrolled to the right and the end of the command is hidden.

```
RP/0/0/CPU0:router(config)# ipv4 access-list 101 permit tcp 172.31.134.5 255.255.255.0 17$
```

In the next example, the Right Arrow key has been used to scroll to the right. Notice that dollar sign symbols appear at both ends of the line, which indicates that command information is hidden from the beginning and end of the command.

```
RP/0/0/CPU0:router(config)# $ccess-list 101 permit tcp 172.31.134.5 255.255.255.0 172.31.$
```

By default, the Cisco IOS XR software uses a terminal screen 80 columns wide. To adjust for a different screen width, use the **terminal width** command in EXEC mode.

Use line wrapping with the command history feature to recall and modify previous complex command entries.

Displaying System Information with show Commands

The **show** commands display information about the system and its configuration. The following sections describe some common **show** commands and provide techniques to manage the output from those commands:

Common show Commands

Table 5-2 shows some of the most common **show** commands.

Table 5-2 Common show Commands in Cisco IOS XR Software

Command	Description	Command Mode
show version	Displays system information.	EXEC or administration EXEC mode
show configuration	Displays the uncommitted configuration changes made during a configuration session.	Global or administration configuration mode
show running-config (EXEC or global configuration mode)	Displays the current running configuration for the SDR to which you are connected.	EXEC or global configuration mode
show running-config (administration EXEC or administration configuration mode)	Displays the current running configuration that applies to the entire router	administration EXEC or administration configuration mode
show tech-support	Collects a large amount of system information for troubleshooting. You can provide this output to technical support representatives when reporting a problem.	EXEC or administration EXEC mode

Table 5-2 Common show Commands in Cisco IOS XR Software (continued)

Command	Description	Command Mode
show platform (EXEC mode)	Displays information about cards and modules assigned to the SDR to which you are connected.	EXEC mode
show platform (administration EXEC mode)	Displays information about all cards and modules in the router.	administration EXEC mode
show environment	Displays hardware information for the system, including fans, LEDs, power supply voltage and current, and temperatures. Enter **show environment ?** to see additional command options.	EXEC mode or administration EXEC mode

For more information on the use of these commands, see the "Related Documents" section on page x.

Browsing Display Output When the --More-- Prompt Appears

When command output requires more than one screen, such as for the **?**, **show**, or **more** commands, the output is presented one screen at a time, and a `--More--` prompt appears at the bottom of the screen.

To display additional command output, do one of the following:

- Press **Return** to display the next line.
- Press **Spacebar** to display the next screen of output.

The following example shows one screen of data and the `--More--` prompt:

```
RP/0/0/CPU0:router# show ?

 MgmtMultilink         Show trace data for the multilink controller component
  aaa                   Show AAA configuration and operational data
  access-lists          Access lists
  address-pool          Local address pool
  adjacency             Adjacency information
  af-ea                 AF-EA Platform details
  aliases               Display alias commands
  app-obj               APP-OBJ Show Commands
  aps                   SONET APS information
  aqsm                  AQSM show commands
  aqsmlib               AQSMLIB show commands
  arm                   IP ARM information
  arp                   ARP show commands
  arp-gmp               ARP show commands
  asic-errors           ASIC error information
  atc                   Attractor Cache related
  atm                   ATM information
  atm-vcm               Show atm_vcm component
  attractor             Show commands for attractor process
  attribute             IM Attributes operations information
  auto-rp               Auto-RP Commands
 --More--
```

Tip If you do not see the `--More--` prompt, try entering a value for the screen length with the **terminal length** command in EXEC mode. Command output is not paused if the **length** value is set to zero. The following example shows how to set the terminal length:

```
RP/0/0/CPU0:router# terminal length 20
```

For information on searching or filtering CLI output, see the "Filtering show Command Output" section on page 5-102.

Halting the Display of Screen Output

To interrupt screen output and terminate a display, press **Ctrl-C**, as shown in the following example:

```
RP/0/0/CPU0:router# show running-config
<Ctrl-C>
```

Redirecting Output to a File

By default, CLI command output appears on the screen. CLI command output can be redirected to a user-specified file by entering a filename and location after the **show** command syntax. The following command syntax is used to redirect output to a file:

show *command* | **file** *filename*

This feature enables you to save any of the **show** command output in a file for further analysis and reference. When you choose to redirect command output, consider the following guidelines:

- If the full path of the file is not specified, the default directory for your account is used. You should always save your target configuration files to this location.
- If the saved output is to be used as a configuration file, the filename should end with the `cfg` suffix for easy identification. This suffix is not required, but can help locate target configuration files. Example: `myconfig.cfg`

In the following example, a target configuration file is saved to the default user directory:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# show configure | file disk0:myconfig.cfg
RP/0/0/CPU0:router(config)# abort
RP/0/0/CPU0:router#
```

Narrowing Output from Large Configurations

Displaying a large running configuration can produce thousands of lines of output. To limit the output of a **show** command to only the items you want to view, use the procedures in the following sections:

- Limiting show Command Output to a Specific Feature or Interface, page 5-100
- Using Wildcards to Display All Instances of an Interface, page 5-101

Limiting show Command Output to a Specific Feature or Interface

Entering keywords and arguments in the **show** command limits the **show** output to a specific feature or interface.

In the following example, only information about the static IP route configuration appears:

```
RP/0/0/CPU0:router# show running-config router static
```

```
router static
 address-family ipv4 unicast
  0.0.0.0/0 10.21.0.1
  0.0.0.0/0 pos0/1/0/1 10.21.0.1
 !
!
```

In the following example, the configuration for a specific interface appears:

```
RP/0/0/CPU0:router# show running-config interface POS 0/1/0/1

interface pos0/1/0/1
 ipv4 address 10.21.54.31 255.255.0.0
 !
```

Using Wildcards to Display All Instances of an Interface

To display the configuration for all instances, enter the asterisk (*) wildcard character.

Note For more information, see the "Using Wildcards to Identify Interfaces in show Commands" section on page 5-109.

In the following example, a configuration for all Packet-over-SONET/SDH (POS) interfaces is displayed:

```
RP/0/0/CPU0:router# show running-config interface pos *

interface POS0/1/0/0
 ipv4 address 10.2.3.4 255.255.255.0
 pos
  crc 32
 !
 shutdown
 keepalive disable
!
interface POS0/1/0/1
 ipv4 address 10.2.3.5 255.255.255.0
 pos
  crc 32
 !
 shutdown
 keepalive disable
!
interface POS0/1/0/2
 ipv4 address 10.2.3.6 255.255.255.0
 pos
  crc 32
 !
 shutdown
 keepalive disable
!
interface POS0/1/0/3
 ipv4 address 10.2.3.7 255.255.255.0
 pos
  crc 32
 !
 shutdown
 keepalive disable
!
```

```
--More--
```

Filtering show Command Output

Output from the **show** commands can generate a large amount of data. To display only a subset of information, enter the "pipe" character (|) followed by a keyword (**begin**, **include**, **exclude**, or **file**) and a regular expression. Table 5-3 shows the filtering options for the **show** command.

Table 5-3 show Command Filter Options

Command	Description
show *command* | **begin** *regular-expression*	Begins unfiltered output of the **show** command with the first line that contains the regular expression.
show *command* | **exclude** *regular-expression*	Displays output lines that do not contain the regular expression.
show *command* | **include** *regular-expression*	Displays output lines that contain the regular expression.
show *command* | **file** *device0:path/file*	Saves output of the **show** command to the specified file on the specified device.
show *command* | **utility** *name*	Displays a set of UNIX utilities: • **cut**—Cuts characters or lines from the output displayed from standard input or a file. • **egrep**—Searches a file using full regular expressions. • **fgrep**—Searches a file for a fixed character string. • **head**—Copies bytes or lines at the beginning of the output displayed from standard input or a file. • **less**—Displays the output of a file in a page-by-page manner. • **sort**—Sorts, merges, or sequence-checks the output displayed from standard input or a file. • **tail**—Copies the end portion of the output displayed from standard input or a file. • **uniq**—Displays or removes repeated lines in a file. • **wc**—Count words, lines, or bytes in a file. • **xargs**—Invokes a program from one or more argument lists.

In the following example, the **show interface** command includes only lines in which the expression "protocol" appears:

```
RP/0/0/CPU0:router# show interface | include protocol

Null0 is up, line protocol is up
0 drops for unrecognized upper-level protocol
POS0/2/0/0 is administratively down, line protocol is administratively down
0 drops for unrecognized upper-level protocol
POS0/2/0/1 is administratively down, line protocol is administratively down
0 drops for unrecognized upper-level protocol
POS0/2/0/2 is administratively down, line protocol is administratively down
0 drops for unrecognized upper-level protocol
POS0/2/0/3 is administratively down, line protocol is administratively down
```

```
0 drops for unrecognized upper-level protocol
MgmtEthernet0/0/CPU0/0 is administratively down, line protocol is administratively
down
MgmtEthernet0/0/CPU0/0 is administratively down, line protocol is administratively
down
0 drops for unrecognized upper-level protocol
```

Note Filtering is available for submodes, complete commands, and anywhere that <cr> appears in the "**?**" output.

Adding a Filter at the --More-- Prompt

You can specify a filter at the `--More--` prompt of a **show** command output by entering a forward slash (**/**) followed by a regular expression. The filter remains active until the command output finishes or is interrupted (using **Ctrl-Z** or **Ctrl-C**). The following rules apply to this technique:

- If a filter is specified at the original command or previous `--More--` prompt, a second filter cannot be applied.
- The use of the **begin** keyword does not constitute a filter.
- The minus sign (–) preceding a regular expression displays output lines that do not contain the regular expression.
- The plus sign (+) preceding a regular expression displays output lines that contain the regular expression.

In the following example, the user adds a filter at the `--More--` prompt to show only the lines in the remaining output that contain the regular expression "ip".

```
RP/0/0/CPU0:router# show configuration running | begin line

Building configuration...
line console
 exec-timeout 120 120
!
logging trap
--More--
/ip
filtering...
ip route 0.0.0.0 255.255.0.0 pos0/2/0/0
interface pos0/2/0/0
 ip address 172.19.73.215 255.255.0.0
end
```

Tip On most systems, **Ctrl-Z** can be entered at any time to interrupt the output and return to EXEC mode.

For more information, see Appendix A, "Understanding Regular Expressions, Special Characters, and Patterns."

Multipipe Support

The multipipe feature supports the multiple pipes on the CLI. With this feature, the output can be processed by an enhanced utility set. Using various combination of utilities, it is possible to gather, filter, and format the output of any **show** command. An arbitrary limit of eight pipes is supported on CLI with this limit superseded by the limit of characters that can be typed on the single line (1024) if the individual commands specified with pipes are long enough.

In addition, if you want to give pipe character (|) as a pattern, you must give it in double quotes. For example:

```
RP/0/0/CPU0:single8-hfr# show running-config|include "bgp|ospf"|file disk0:/usr/a.log
```

show parser dump command

The **show parser dump** command displays the CLI syntax options for a specific submode.

It is a utility that dumps the parser commands supported on the router and a tool that displays line-by-line commands available in a submode. The command is available in every mode and it shows the command set available for that mode. This is a very handy tool for collecting the CLI commands for a mode.

The **show parser dump** command supports a filter. For example, an initial portion of the command can be specified and the command set matching to that portion can be displayed.

```
RP/0/0/CPU0:router(config-un)# show parser dump

show
show configuration merge
show configuration running sanitized desanitize rpl
show configuration running sanitized
show configuration running
show configuration
show configuration failed noerrors
show configuration failed
show configuration failed load
show running-config
show running-config sanitized desanitize rpl
show running-config sanitized
show running-config submode
show parser dump
show history detail
show history
pwd
exit
```

Accessing Admin Commands from Secure Domain Router Mode

You can access admin commands from secure domain router mode by prefixing the **Admin** keyword. Switching to admin mode is not required. For example:

```
RP/0/0/CPU0:router# admin install add
tftp://223.255.254.254/muck/username/38ws/hfr-mpls-p.pie sync active
```

In the preceding example the **install** command is an admin mode command that you can run from SDR by prefixing **admin** keyword.

Location Keyword for the File Command

Specify the location of the media (as specified, disk0) where the file needs to be stored. This option is available only for the disk or any media storage available on different nodes of the router.

If you have a media (disk0: disk1:), it is provided with an additional location keyword. This option displays all the nodes where the media is present.

```
RP/0/0/CPU0:router# sh logging | file disk0:/log-file location ?

0/0/cpu0 Fully qualified location specification
0/1/cpu0 Fully qualified location specification
```

Note The **location** keyword must be available only for the disk or any media storage available on RP. Network files do not require this keyword.

vty / Console Timestamp

This feature enables the timestamp to be set to *On* by default for each EXEC or admin EXEC command. Previously, the default setting for the time stamp was disabled.

The following command disables the timestamp:

```
RP/0/0/CPU0:router(config)# line console timestamp disable
```

The following command enables the timestamp:

```
RP/0/0/CPU0:router(config)# no line console timestamp disable
```

However, the previous command to enable the timestamp is still available.

Displaying Interfaces by Slot Order

This feature lets you display physical interfaces in a sequence of slots for a specific rack. This provides an easy way to determine if the interfaces are configured on a specific slot. Previously, the physical interfaces were displayed by interface types.

To display the interfaces by slot order, you need to configure the **configuration display interface slot-order** command at the global configuration mode.

```
RP/0/0/CPU0:router# configure terminal
RP/0/0/CPU0:router(config)# configuration display interface slot-order
RP/0/0/CPU0:router(config)# commit
RP/0/0/CPU0:router(config)# end
```

This command enables the display of physical interfaces by slot-order:

```
RP/0/0/CPU0:router# show running-config
Building configuration...
!! IOS XR Configuration 4.1.0.29I
!! Last configuration change at Mon Mar 21 06:35:17 2011 by lab
!
service configuration display slot-order
interface MgmtEth0/0/CPU0/0
 ipv4 address 12.29.38.6 255.255.0.0
!
interface MgmtEth0/0/CPU0/1
 shutdown
!

interface POS0/2/0/0
 shutdown
```

```
!
interface POS0/2/0/1
 shutdown
!
interface GigabitEthernet0/3/0/0
 shutdown
!
interface GigabitEthernet0/3/0/1
 shutdown
!

interface POS0/4/0/0
 shutdown
!
interface POS0/4/0/1
 shutdown
```

Note The **configuration display interface slot-order** command is supported only in the SDR configuration mode.

Displaying Unconfigured Interfaces

This feature lets you display the list of all physical interfaces, even if these interfaces are not configured. You can use the **show running-config all-interfaces** command to display all unconfigured interfaces. Previously, the **show running-config** command displayed only the running configuration of the system--any feature not configured explicitly by the user (or operating in default mode) would not have any evidence in the output of the **show running-config** command.

```
RP/0/0/CPU0:router# show running-config all-interfaces
Sun Jun 13 21:44:46.769 DST
Building configuration...
!! IOS XR Configuration 4.1.0.29I
!! Last configuration change at Mon Mar 21 06:35:17 2011 by lab
!
hostname Router
interface MgmtEth0/0/CPU0/0
 ipv4 address 12.29.38.6 255.255.0.0
!
interface MgmtEth0/0/CPU0/1
 shutdown
!
interface POS0/2/0/0
!
interface POS0/2/0/1
!
router static
 address-family ipv4 unicast
   0.0.0.0/0 12.29.0.1
 !
!
```

Notice that the POS interfaces have no configurations but they are still shown in the output of the command.

This option is not applicable to other variants of show configuration commands like the following:

- **show configuration**
- **show configuration commit changes**
- **show configuration rollback changes**
- **show configuration failed**
- **show configuration persistent**

Displaying Subnet Mask in CIDR Format

This feature displays IPv4 address subnet mask in Classless Interdomain Routing (CIDR) format instead of decimal format. The change of format for all show commands may cause backward compatibility issues. To overcome this problem, the **ipv4 netmask-format hit-count** command has been implemented in the IP/CLI component, which maintains the common infrastructure specific to IP related CLIs.

To display the subnet in a prefix length format, you need to configure the **ipv4 netmask-format hit-count** command at the global configuration mode:

```
RP/0/0/CPU0:router# configure terminal
RP/0/0/CPU0:router(config)# ipv4 netmask-format bit-count
RP/0/0/CPU0:router(config)# commit
RP/0/0/CPU0:router(config)# end
```

After this command has been configured, the output of the show command forcefully displays the subnet mask in a prefix length format. Also, you can disable the command by using the **no** form of the command.

```
RP/0/0/CPU0:router# no ipv4 netmask-format bit-count
RP/0/0/CPU0:router#
```


Note This **ipv4 netmask-format hit-count** command is supported only in the SDR configuration mode.

The following example shows the output of a **show running-config** command after the **ipv4 netmask-format bit-count** command has been configured:

```
RP/0/0/CPU0:router# show running-config interface mgmtEth 0/RP0/CPU0/0
Mon May 31 23:48:17.453 DST
interface MgmtEth0/RP0/CPU0/0
 description Connected to Lab LAN
 ipv4 address 172.29.52.70 255.255.255.0
!
```

Wildcards, Templates, and Aliases

This section contains the following topics:

Using Wildcards to Identify Interfaces in show Commands

Wildcards (*) identify a group of interfaces in **show** commands. Table 5-4 provides examples of wildcard usage to identify a group of interfaces.

Table 5-4 Examples of Wildcard Usage

Wildcard Syntax	Description
*	Specifies all interfaces
pos*	Specifies all POS interfaces in the system
pos0/1/*	Specifies all POS interfaces in rack 0, slot 1
pos0/3/4*	Specifies all subinterfaces for POS0/3/4

Note The wildcard (*) must be the last character in the interface name.

Example

The following example shows how the configuration for all POS interfaces in rack 0, slot 1 is displayed:

```
RP/0/0/CPU0:router:router# show running-config interface pos0/1/*

interface POS0/1/0/0
 ipv4 address 10.2.3.4 255.255.255.0
 pos
  crc 32
 !
 keepalive disable
interface POS0/1/0/1
 ipv4 address 10.2.3.5 255.255.255.0
 pos
  crc 32
 !
 keepalive disable
interface POS0/1/0/2
 ipv4 address 10.2.3.6 255.255.255.0
 pos
  crc 32
 !
 keepalive disable
interface POS0/1/0/3
 ipv4 address 10.2.3.7 255.255.255.0
 pos
```

```
 crc 32
!
keepalive disable

--More--
```

The following example shows how the state of all POS interfaces is displayed:

```
RP/0/0/CPU0:router# show interfaces pos* brief

                Intf        Intf         LineP              Encap  MTU       BW
                Name        State        State               Type (byte)    (Kbps)
--------------------------------------------------------------------------------
           PO0/1/0/0          up           up                 HDLC  4474   2488320
           PO0/1/0/1          up           up                 HDLC  4474   2488320
           PO0/1/0/2          up           up                 HDLC  4474   2488320
           PO0/1/0/3          up           up                 HDLC  4474   2488320
           PO0/1/0/4          up           up                 HDLC  4474   2488320
           PO0/1/0/5          up           up                 HDLC  4474   2488320
           PO0/1/0/6          up           up                 HDLC  4474   2488320
           PO0/1/0/7          up           up                 HDLC  4474   2488320
           PO0/1/0/8          up           up                 HDLC  4474   2488320
           PO0/1/0/9          up           up                 HDLC  4474   2488320
          PO0/1/0/10          up           up                 HDLC  4474   2488320
          PO0/1/0/11          up           up                 HDLC  4474   2488320
          PO0/1/0/12          up           up                 HDLC  4474   2488320
          PO0/1/0/13          up           up                 HDLC  4474   2488320
          PO0/1/0/14          up           up                 HDLC  4474   2488320
          PO0/1/0/15          up           up                 HDLC  4474   2488320
```

Creating Configuration Templates

Configuration templates allow you to create a name that represents a group of configuration commands. After a template is defined, it can be applied to interfaces by you or other users. As networks scale to large numbers of nodes and ports, the ability to configure multiple ports quickly using templates can greatly reduce the time it takes to configure interfaces.

The two primary steps in working with templates are creating templates and applying templates. The following procedure describes how to create a configuration template.

SUMMARY STEPS

1. **configure**
2. **template** *template-name* [(**$***parameter* **$***parameter...*)] [*config-commands*]
3. Enter the template commands.
4. **end-template**
5. **commit**
6. **show running-config template** *template-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `Router# configure`	Enters global configuration mode.
Step 2	**template** *template-name* [(**$***parameter* **$***parameter...*)] [*config-commands*] **Example:** `RP/0/0/CPU0:router(config)# template tmplt_1`	Enters template configuration mode and creates a template. • *template-name*—Unique name for the template to be applied to the running configuration. • (Optional) *parameter*—Actual values of the variables specified in the template definition. Up to five parameters can be specified within parentheses. Each parameter must begin with the **$** character. Templates can be created with or without parameters. • (Optional) *config-commands*—Global configuration commands to be added to the template definition. Any name in a command (such as the server name, group name, and so on) can be parameterized. This means that those parameters can be used in the template commands (starting with $) and replaced with real arguments when applied. • To remove the template, use the **no** form of this command.
Step 3	Enter the template commands. **Example:** `RP/0/0/CPU0:router(config-TPL)# hostname test`	Defines the template commands.
Step 4	**end-template** **Example:** `RP/0/0/CPU0:router(config-TPL)# end-template`	Ends the template definition session and exits template configuration mode. • When you end the template session, you are returned to global configuration mode.
Step 5	**commit** **Example:** `RP/0/0/CPU0:router(config-TPL)# commit`	Applies the target configuration commands to the running configuration.
Step 6	**show running-config template** *template-name* **Example:** `RP/0/0/CPU0:router# show running-config template tmplt_1`	Displays the details of the template.

Examples

The following example shows how a simple template is defined. The template contents are then displayed with the **show running-config template** *template-name* command:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# template jbtest
RP/0/0/CPU0:router(config-TPL)# hostname test
RP/0/0/CPU0:router(config-TPL)# end-template
RP/0/0/CPU0:router(config)# commit
RP/0/0/CPU0:router(config)# show running-config template jbtest

template jbtest
 hostname test
end-template
```

In the next example, a template is defined, and the template requires a parameter. The template contents are then displayed with the **show running-config template** *template-name* command:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# template test2 (hostname)
RP/0/0/CPU0:router(config-TPL)# hostname $hostname
RP/0/0/CPU0:router(config-TPL)# end-template
RP/0/0/CPU0:router(config)# commit
RP/0/0/CPU0:router(config)# show running-config template test2

template test2 (hostname)
 hostname $hostname
end-template
```

Applying Configuration Templates

To apply a template, enter the **apply-template** *template-name* [(*parameter*)] command in global configuration mode and consider the following guidelines:

- Only one template can be applied at a time.
- If the same template is applied multiple times, the most recent application overwrites the previous ones.
- Provide the exact number of parameters for the template.
- Templates are applied as a “best effort” operation; only valid changes are committed. If any command in the template fails, that command is discarded.
- After a template is applied, the **show configuration** command displays the target configuration changes. The target configuration must be committed (with the **commit** command) to become part of the running configuration.

Examples

In the following example, a simple template is defined. The template contents are then displayed with the **show running-config template** *template-name* command:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# apply-template jbtest
RP/0/0/CPU0:router(config)# show configuration
Building configuration...
hostname test
end
```

In the next example, a template with one parameter is applied and the **show configuration** command displays the result:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# apply-template test2 (router)
RP/0/0/CPU0:router(config)# show configuration

Building configuration...
hostname router
end
```

Aliases

With the Cisco IOS XR software, you can define command-line aliases for any physical or logical entity in a router. After you define the alias, it is used in the CLI to reference the real entity.

To create a command alias, enter the **alias** command in global configuration or administration configuration mode:

alias *alias-name* [(*parameter1 parameter2...*)] *command-syntax* [$*parameter1*] *command-syntax* [$*parameter2*]

Table 5-5 defines the **alias** command syntax.

Table 5-5 alias Command Syntax

Syntax	Specifies that the Alias Is Created for
alias-name	Name of the command alias. An alias name can be a single word or multiple words joined by a dash (–) delimiter.
command-syntax	Original command syntax. Valid abbreviations of the original command syntax can be entered for the *command-syntax* argument.
(*parameterx*)	Argument or keyword that belongs to the command you specified for the *command-syntax* argument. When the parameter is entered in parenthesis after the alias name, the alias requires a parameter name. To associate the parameter with a command within the alias, enter the $ character preceding the parameter name.

Multiple commands can be supported under a single command alias, and multiple variables can be supported for each command. If multiple commands are specified under a single alias, each command is executed in the order in which it is listed in the **alias** command.

In the following example, an alias named *my-cookie* is created for the Management Ethernet interface, and then the new alias is specified to enter interface configuration mode:

```
RP/0/0/CPU0:router(config)# alias my-cookie mgmtEth 0/0/CPU0/0
RP/0/0/CPU0:router(config)# interface my-cookie
RP/0/0/CPU0:router(config)# interface mgmtEth 0/0/CPU0/0
RP/0/0/CPU0:router(config-if)#
```

After you enter a command with an alias, the router displays the command you entered with the alias value so that you can verify that alias value.

To delete a specific alias, enter the **no** form of the **alias** command with the alias name.

Keystrokes Used as Command Aliases

The system can be configured to recognize particular keystrokes (key combination or sequence) as command aliases. In other words, a keystroke can be set as a shortcut for executing a command. To enable the system to interpret a keystroke as a command, use the **Ctrl-V** or **Esc**, **Q** key combination before entering the command sequence.

Command History

The Cisco IOS XR software lets you display a history of the most recently entered and deleted commands. You can also redisplay the command line while a console message is being shown. The following sections describe the command history functionality:

Note To roll back to a previously committed configuration, see Managing Configuration History and Rollback.

Displaying Previously Entered Commands

The Cisco IOS XR software records the ten most recent commands issued from the command line in its history buffer. This feature is particularly useful for recalling long or complex commands or entries, including access lists.

To display commands from the history buffer, enter the **show history** command as follows:

```
RP/0/0/CPU0:router# show history

  show configuration history commit
  show configuration commit list
show config commit changes 1000000001
  show history
```

Recalling Previously Entered Commands

The Cisco IOS XR software records the ten most recent commands issued from the command line in its history buffer. This feature is particularly useful for recalling long or complex commands or entries, including access lists.

Table 5-6 lists the commands or key strokes to use to recall commands from the history buffer.

Table 5-6 Command History

Command or Key Combination	Purpose
Ctrl-P or the **Up Arrow** key	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or the **Down Arrow** key	Returns to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the **Up Arrow** key. Repeat the key sequence to recall successively more recent commands.

Recalling Deleted Entries

The Cisco IOS XR CLI also stores deleted commands or keywords in a history buffer. The buffer stores the last ten items that have been deleted using **Ctrl-K**, **Ctrl-U**, or **Ctrl-X**. Individual characters deleted using **Backspace** or **Ctrl-D** are not stored.

Table 5-7 identifies the keystroke combinations used to recall deleted entries to the command line.

Table 5-7 Keystroke Combinations to Recall Deleted Entries

Command or Key Combination	Recalls
Ctrl-Y	Most recent entry in the buffer (press the keys simultaneously).
Esc, **y**	Previous entry in the history buffer (press the keys sequentially).

Note The **Esc**, **y** key sequence does not function unless the **Ctrl-Y** key combination is pressed first. If the **Esc, y** is pressed more than ten times, the history cycles back to the most recent entry in the buffer.

Redisplaying the Command Line

If the system sends a message to the screen while a command is being entered, the current command-line entry can be redisplayed using the **Ctrl-L** or **Ctrl-R** key combination.

Displaying Persistent CLI History

The Cisco IOS XR maintains the history buffer of CLI commands persistently across user sessions, router switchover, and router reloads. This buffer not only provides a log of commands entered by various users, but also lets you trace the activity of active users if the threshold limit of CPU usage is exceeded. This command is useful for troubleshooting purposes.

To display the history of events corresponding to the CLI session open events, enter the **show cli history brief location** command at the EXEC mode as follows:

```
RP/0/0/CPU0:router# show cli history brief location 0/RP0/CPU0

No.        Username            Line            IPAddress     Client            t
--- --------------- --------------- -------------------- ---------- ------------
  1               -               -                    -          -  Thu Jun 11e
```

```
  2               -               -                   -          -  Thu Jun 11e
  3               -               -                   -          -  Thu Jun 11e
  4        jhensper   con0_RP0_CPU0                   -       exec  Thu Jun 11n
  5        jhensper   con0_RP0_CPU0                   -  adminexec  Thu Jun 11n
```

To display the history of commands from each session along with user name, enter the **show cli history detail location** command at the EXEC mode as follows:

```
RP/0/0/CPU0:router# show cli history detail location 0/RP0/CPU0
Sun Jun 13 21:52:10.219 DST
 No.  Username        Line         Client    Time            Command
---- --------------- --------------- ------------------- ----------------------------
   1   lab      vty0       adminexec  Mon May 31 22:10:23.156 PST show configuration
commit list
   2   lab      vty0            adminexec  Mon May 31 22:10:31.352 PST exit
   3   lab      vty0                 exec  Mon May 31 22:10:45.627 PST admin
   4   lab      vty1                 exec  Mon May 31 22:12:03.853 PST configure
   5   lab      vty1               config  Mon May 31 22:12:06.463 PST mpls traffic-eng
```

The **detail** option displays the commands from each session along with user name and vty id so that commands issued from a session can be related with the session history displayed in the **brief** option.

Note The default size is 500 for the brief option of the command. The default size is 1000 for the detail option of the command.

Key Combinations

The following sections provide information on key combinations:

Key Combinations to Move the Cursor

Table 5-8 shows the key combinations or sequences you can use to move the cursor around on the command line to make corrections or changes. When you use cursor control keys, consider the following guidelines:

- Ctrl indicates the Control key, which must be pressed simultaneously with its associated letter key.
- Esc indicates the Escape key, which must be pressed first, followed by its associated letter key.
- Keys are not case sensitive.

Table 5-8 Key Combinations Used to Move the Cursor

Keystrokes	Function	Moves the Cursor
Left Arrow or **Ctrl-B**	Back character	One character to the left. When you enter a command that extends beyond a single line, you can press the **Left Arrow** or **Ctrl-B** keys repeatedly to scroll back toward the system prompt and verify the beginning of the command entry, or you can press the **Ctrl-A** key combination.
Right Arrow or **Ctrl-F**	Forward character	One character to the right.
Esc, **b**	Back word	Back one word.
Esc, **f**	Forward word	Forward one word.
Ctrl-A	Beginning of line	To the beginning of the line.
Ctrl-E	End of line	To the end of the command line.

Keystrokes to Control Capitalization

Letters can be uppercase or lowercase using simple key sequences. Table 5-9 describes the keystroke combinations used to control capitalization.

Note Cisco IOS XR commands are generally case insensitive and typically all in lowercase.

Table 5-9 Keystrokes Used to Control Capitalization

Keystrokes	Purpose
Esc, **c**	Makes the letter at the cursor uppercase.
Esc, **l**	Changes the word at the cursor to lowercase.
Esc, **u**	Makes letters from the cursor to the end of the word uppercase.

Keystrokes to Delete CLI Entries

Table 5-10 describes the keystrokes used to delete command-line entries.

Table 5-10 Keystrokes for Deleting Entries

Keystrokes	Deletes
Delete or **Backspace**	Character to the left of the cursor.
Ctrl-D	Character at the cursor.
Ctrl-K	All characters from the cursor to the end of the command line.
Ctrl-U or **Ctrl-X**	All characters from the cursor to the beginning of the command line.
Ctrl-W	Word to the left of the cursor.
Esc, **d**	From the cursor to the end of the word.

Transposing Mistyped Characters

To transpose mistyped characters, use the **Ctrl-T** key combination.

CHAPTER 6

Troubleshooting the Cisco IOS XR Software

This chapter describes the tools and procedures used to identify the source of hardware and software problems. This chapter also provides instructions on gathering data for further analysis by Cisco customer support representatives.

Contents

Additional Sources of Information

For additional information on troubleshooting, see the following sources:

- If the Cisco IOS XR software does not start and display the EXEC mode prompt, see *Cisco IOS XR ROM Monitor Guidefor the Cisco XR 12000 Series Router.*
- Technical Assistance Center (TAC) home page, containing 30,000 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/public/support/tac/home.shtml
- "Related Documents" section on page x.

Basic Troubleshooting Commands

The following sections describe some basic techniques used to determine connectivity to another device and display information on the configuration and operation of a router.

- Using debug Commands, page 6-123

Using show Commands to Display System Status and Configuration

Use the **show** commands to check the status of various Cisco IOS XR software subsystems and services. Table 6-1 lists some of the common **show** commands.

To display a complete list of the available show commands, enter the **show ?** command to access the on-screen help system.

Note Different **show** commands are available in different command modes, and the same **show** command can show different results in different command modes.

Table 6-1 Common show Commands in Cisco IOS XR Software

Command	Description
show variables boot (EXEC and administration EXEC modes)	Displays the boot variables.
show configuration (Global configuration and administration configuration modes)	Displays the uncommitted configuration changes made during a configuration session. This command can be entered in any configuration mode.
show context (and show exception) (EXEC and administration EXEC modes)	Displays context information about all recent reloads.
show controller (Administration EXEC mode)	Displays hardware controller information.
show controllers (EXEC mode)	Displays hardware controller information.
show debug (EXEC and administration EXEC modes)	Displays debug flags enabled from the current terminal.
show environment [*options*] (EXEC and administration EXEC modes)	Displays hardware information for the physical components and systems, including fans, LEDs, power supply voltage and current information, and temperatures. To view the command options, enter the **show environment ?** command.
show exception (EXEC and administration EXEC modes)	Displays all exception dump configurations.
show install (EXEC and administration EXEC modes)	Displays installed and active software packages.
show interfaces (EXEC mode)	Displays interface status and configuration.

Table 6-1 Common show Commands in Cisco IOS XR Software (continued)

Command	Description
show logging (EXEC and administration EXEC modes)	Displays the contents of logging buffers.
show memory (EXEC and administration EXEC modes)	Displays memory statistics.
show platform (EXEC and administration EXEC modes)	Displays information about node status on the router. To display the nodes assigned to an SDR, enter this command in EXEC mode. To display all the nodes in a router, enter this command in administration EXEC mode.
show processes blocked (EXEC and administration EXEC modes)	Displays blocked processes.
show redundancy (EXEC and administration EXEC modes)	Display the status of the primary (active) RP[1] and the standby (redundant) RP.
show running-config [*command*] (EXEC and administration EXEC modes)	Displays the current running configuration.
show tech-support (EXEC and administration EXEC modes)	Collects a large amount of system information for troubleshooting. The output should be provided to technical support representatives when a problem is reported. Because of the impact the command can have on a running system, it is reserved for users assigned to the cisco-support task ID.
show user [**group** | **tasks** | **all**] (EXEC mode)	Displays the username for the current logged-in user. Use this command to also display the groups and associated task IDs assigned to the account.
show version (EXEC and administration EXEC modes)	Displays basic system information.

1. RP stands for Route Processor

Using the ping Command

Use the **ping** command to diagnose network connectivity. In EXEC mode, enter a hostname or an IP address as an argument to this command. In administration EXEC mode, you can use the fabric or the control Ethernet network (in a multishelf system) to ping other nodes.

The **ping** command sends an echo request packet to a destination, then awaits a reply. Ping output can help you evaluate path-to-destination reliability, delays over the path, and whether the destination can be reached or is functioning.

Each exclamation point (!) indicates receipt of a reply. A period (.) indicates the network server timed out while waiting for a reply. Other characters may appear in the ping output display, depending on the protocol type.

Examples

The following example shows a successful ping attempt:

```
RP/0/0/CPU0:router# ping 10.233.233.233

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.233.233.233, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/7 ms
```

In the next example, an unsuccessful ping attempt is shown:

```
RP/0/0/CPU0:router# ping 10.1.1.1

Mon May 31 23:53:30.820 DST
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

Using the traceroute Command

Use the **traceroute** command in EXEC mode to discover the routes that packets take when traveling to their destination. Enter a hostname or an IP address as an argument to this command.

This command works by taking advantage of the error messages generated by routers when a datagram exceeds its time-to-live (TTL) value.

The **traceroute** command starts by sending probe datagrams with a TTL value of 1, causing the first router to discard the probe datagram and send back an error message. The **traceroute** command sends several probes at each TTL level and displays the round-trip time for each.

The **traceroute** command sends one probe at a time. Each outgoing packet may result in one or two error messages. A *time exceeded* error message indicates that an intermediate router has seen and discarded the probe. A *destination unreachable* error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet. If the timer times out before a response comes in, the **traceroute** command prints an asterisk (*).

The **traceroute** command terminates when the destination responds, the maximum TTL is exceeded, or the user interrupts the trace with the escape sequence.

Examples

The following example shows how the route for an IP address appears:

```
RP/0/0/CPU0:router# traceroute 10.233.233.233

Mon May 31 23:55:23.034 DST

Type escape sequence to abort.
Tracing the route to 10.233.233.233

 1  ce28 (172.29.52.1) 2 msec  1 msec  0 msec
 2  172.24.114.17 0 msec  0 msec  0 msec
 3  172.24.114.17 !A  *  !A
```

Using debug Commands

Debug commands are used to diagnose and resolve network problems. Use **debug** commands to troubleshoot specific problems or during troubleshooting sessions.

Use **debug** commands to turn on or off debugging for a specific service or subsystem. When debugging is turned on for a service, a debug message is generated each time the debugging code section is entered.

The following sections provide information on debugging:

- Displaying a List of Debug Features, page 6-123
- Enabling Debugging for a Feature, page 6-125
- Disabling Debugging for a Service, page 6-126
- Displaying Debugging Status, page 6-125
- Disabling Debugging for All Services Started at the Active Terminal Session, page 6-126
- Disabling Debugging for All Services Started at All Terminal Sessions, page 6-126

Caution Debug commands can generate a large amount of output and can render the system unusable. Use the **debug** commands to troubleshoot specific problems or during specific troubleshooting sessions on systems that are not in production.

Displaying a List of Debug Features

To display a list of the available debug features, enter the debug mode and enter a **?** for on-screen help. The set of debug mode features is different in EXEC and administration EXEC modes. In the following example, EXEC mode is the entry point to debug mode:

```
RP/0/0/CPU0:router# debug
RP/0/0/CPU0:router(debug)# ?

  MgmtMultilink      MgmtMultilink controller debugging
  aaa                AAA Authentication, Authorization and Accounting
  access-list        Debug ethernet-services access-lists
  accounting         Turn on Accounting debug
  adjacency          platform AIB information
  afmon-ea           debug afmon-ea services
  afmon-lib          debug afmon client library specific function calls
  afmon-ma           Debug afmon-ma services
  aib                AIB information
```

```
 aipc-em             Debug aipc ethernet support
 alarm-location      Alarm Location library debugging
 alarm-logger        Turn on alarm debugging
 ancp                ANCP Debug Information
 ap                  Address Pool Debugs
 app                 Address Pool Proxy Debugs
 app-obj             Debug app-obj services
 aps                 Address Pool Proxy Debugs
 arm                 IP Address Repository Manager
 arp                 IP ARP transactions
 arp-gmp             ARP Global Management Process debugging
 async               async messaging information
 atm                 ATM debugging
--More--
```

In the next example, administration EXEC mode is the entry point to debug mode:

```
RP/0/0/CPU0:router# admin
RP/0/0/CPU0:router(admin)# debug
RP/0/0/CPU0:router(admin-debug)# ?

caibist      CAI BIST debugging
  cctl         Chassis control driver process debug
  cetftp       Control ethernet TFTP (CE-TFTP) server process debug
  cih          Debug CAI CIH
  cpuctrl      Configure Cpuctrl debug settings
  describe     Describe a command without taking real actions
  diagnostic   Diagnostic debugging
  dsc          dsc debug: all, fsm, table, cfg, and api
  dumper       Admin Debug Dumper
  ethernet     Ethernet debug commands
  exit         Exit from this submode
  fabric       Fabric debugging
  fabricq      Debug Fabric Queue Manager
  fia          Debug the Fabric Interface ASIC (FIA) driver
  gsp          Admin Debug gsp
  i2c-ctrl     Debug the functionality of I2C Control
  ingressq     Debug Ingress Queue Manager
  install      Install debug information
  inv          Inventory manager process debug
  invd         Inventory debug: all, trap, dll mem
  invmgr       Inventory Manager client API interface debug
  no           Disable debugging functions
  obfl         OBFL related admin debugs
 --More--
```

Enabling Debugging for a Feature

To enable debugging for a feature, type enter the **debug** command in EXEC or administration EXEC mode and then enable the feature for debugging. For example:

```
RP/0/0/CPU0:router# debug
RP/0/0/CPU0:router(debug)# aaa all
RP/0/0/CPU0:router(debug)# exit
```

You can also enter the complete command from EXEC mode, as shown in the following example:

```
RP/0/0/CPU0:router# debug aaa all
```

Displaying Debugging Status

Enter the **show debug** command to display the debugging features enabled for your terminal session. The terminal session is labeled *tty* and represents your connection to the router through a specific port, which might be the console port, auxiliary port, or Management Ethernet interface. In the following example, the command display indicates that debugging is enabled for two features (AAA and ipv4 io icmp) from a terminal session on the console port of RP1:

```
RP/0/0/CPU0:router# show debug

####  debug flags set from tty 'con0_RP1_CPU0'  ####
aaa all flag is ON
ipv4 io icmp flag is ON

RP/0/0/CPU0:router# no debug aaa all
RP/0/0/CPU0:router# show debug
```

```
####  debug flags set from tty 'con0_RP1_CPU0'  ####
ipv4 io icmp flag is ON
```

On a Cisco XR 12000 Series Router, the slot number of the tty ID is 0 or 1 instead of RP0 or RP1.

Enter the **show debug conditions** command to display the conditional debugging status. For example:

```
RP/0/0/CPU0:router# show debug conditions

####  debug conditions set from tty 'con0_RP1_CPU0'  ####
interface  condition is ON for interface 'POS0/2/0/1'
```

Disabling Debugging for a Service

Use the **no** form of the **debug** command or the **undebug** command to turn off debugging for a service or subsystem.

In the following example, the **no debug** command disables debugging for the AAA feature:

```
RP/0/0/CPU0:router# no debug aaa all
RP/0/0/CPU0:router# show debug

####  debug flags set from tty 'con0_RP1_CPU0'  ####
ipv4 io icmp flag is ON
```

You can also turn off debugging from the undebug mode, as shown in the following example:

```
RP/0/0/CPU0:router# undebug
RP/0/0/CPU0:router(undebug)# aaa all
RP/0/0/CPU0:router(undebug)# exit
```

Disabling Debugging for All Services Started at the Active Terminal Session

Use the **undebug all** or **no debug all** command to turn off all debugging started by the active terminal session. For example, if you enter either of these commands while connected to the router through the console port on the active RP, all debug sessions started from that console port are disabled. In the following example, debugging for all services is disabled and then verified:

```
RP/0/0/CPU0:router# undebug all
RP/0/0/CPU0:router# show debug

No matching debug flags set
```

Disabling Debugging for All Services Started at All Terminal Sessions

Use the **undebug all all-tty** command to turn off debugging for all services that have been started from all terminal sessions. For example if you enter this command while connected to the router through the console port on the active RP, all debug sessions started from all ports are disabled. In the following example, debugging for all services and ports is disabled and then verified:

```
RP/0/0/CPU0:router# undebug all all-tty
RP/0/0/CPU0:router# show debug

No matching debug flags set
```

Configuration Error Messages

The following sections contain information on configuration error messages:

Configuration Failures During a Commit Operation

A target configuration is added to the running configuration of a router when the **commit** command is entered. During this operation, the changes are automatically verified by the other components in the system. If successful, the configuration becomes part of the running configuration. If some configuration items fail, an error message is returned.

To display the configuration items that failed and see the cause of each failure, enter the **show configuration failed** command.

Note The **show configuration failed** command can be entered in either the EXEC mode or any configuration mode. In any mode, the configuration failures from the most recent **commit** operation are displayed.

In the following example, a configuration error occurs when an invalid commit operation is attempted:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# taskgroup bgp
RP/0/0/CPU0:router(config-tg)# description this is an example of an invalid taskgroup
RP/0/0/CPU0:router(config-tg)# commit

% Failed to commit one or more configuration items. Please use 'show configuration failed'
to view the errors
```

To display the configuration items that failed, including a description of the error, enter the **show configuration failed** command:

```
RP/0/0/CPU0:router(config-tg)# show configuration failed

!! CONFIGURATION FAILED DUE TO SEMANTIC ERRORS
taskgroup bgp
!!% Usergroup/Taskgroup names cannot be taskid names
```

You can also display the failed configuration items without the error description by entering the **show configuration failed noerror** command:

```
RP/0/0/CPU0:router(config-tg)# show configuration failed noerror

!! CONFIGURATION FAILED DUE TO SEMANTIC ERRORS
taskgroup bgp
```

Configuration Errors at Startup

Configuration errors that occurred during system startup can be displayed with the **show configuration failed startup** command. For example:

```
RP/0/0/CPU0:router# show configuration failed startup
```

```
!! CONFIGURATION FAILED DUE TO SYNTAX ERRORS
ntp
xml agent corba
http server
```

Memory Warnings in Configuration Sessions

The Cisco IOS XR software automatically monitors and manages the system resources in a router. Under normal operating conditions, memory problems should not occur.

When a low-memory issue does occur, it is often in the form of a low-memory warning during a configuration session. Low-memory conditions can be caused by multiple, large configurations being added to the router at a single time. Users can remove the source of a problem by removing configurations.

The following sections describe the commands used to display memory usage in a router and what to do if a low-memory warning appears:

Understanding Low-Memory Warnings in Configuration Sessions

The Cisco IOS XR software monitors memory usage in the router. If system memory becomes low, an error message appears when you attempt to enter configuration mode.

An “out-of-memory” error message appears during one of the following situations:

- When a user attempts to enter configuration mode.
- During a configuration session when the memory shortage occurs.
- When a user attempts to load a target configuration from a large file that results in a memory shortage.
- During a commit operation that results in the low-memory warning message. The commit operation is denied and only lr-root users can perform commit operations to remove configurations.

Caution Never ignore a low-memory warning. These warnings indicate a memory state that could affect system operations if not addressed.

“WARNING! MEMORY IS IN MINOR STATE”

If the system memory begins to run low, the following minor memory warning appears when you enter a new configuration mode:

```
WARNING! MEMORY IS IN MINOR STATE
```

Although users are allowed to enter configuration mode, they should immediately reduce memory usage using the tools described in the “Removing Configurations to Resolve Low-Memory Warnings” section on page 6-130.

Failure to take action can result in a worsening situation and eventual impact to router operations.

"ERROR! MEMORY IS IN SEVERE (or CRITICAL) STATE"

When the memory is in a severe or critical state, router operation and performance is likely to be affected. Regular users are not allowed to enter configuration mode. Only lr-root owners can enter configuration mode to free memory by removing configurations.

In some situations, the **commit** command is not allowed. Users with lr-root access can still use the **commit force** command to apply configurations that reduce memory usage. Reducing memory usage normally means removing configurations, but a user can also add configurations that reduce memory usage. For example, configuring the **shutdown** command on an interface could cause numerous routes to be purged from Border Gateway Protocol (BGP), the Routing Information Base (RIB), and Forwarding Information Base (FIB) configurations.

Caution The **commit force** command should be used only to apply configurations that reduce memory usage. Adding configurations that increase memory usage could result in serious loss of router operation.

Displaying System Memory Information

To display a high level summary of system memory, enter the **show memory summary** command. Table 6-2 describes the meaning of each heading.

```
RP/0/0/CPU0:router# show memory summary

Tue Jun  1 00:02:03.826 DST
Physical Memory: 4096M total (2020M available)
 Application Memory : 3818M (2020M available)
 Image: 50M (bootram: 50M)
 Reserved: 226M, IOMem: 2028M, flashfsys: 0
 Total shared window: 32M
RP/0/0/CPU0:router#
```

To display general memory usage for the device as a whole and by process, enter the **show memory** command. Table 6-2 describes the meaning of each heading.

```
RP/0/0/CPU0:router# show memory

Tue Jun  1 00:05:44.927 DST
Physical Memory: 4096M total (2020M available)
 Application Memory : 3818M (2020M available)
 Image: 50M (bootram: 50M)
 Reserved: 226M, IOMem: 2028M, flashfsys: 0
 Shared window tunl_gre: 39K
 Shared window statsd_db: 67K
 Shared window l2fib: 323K
 Shared window li: 3K
 Shared window ipv4_fib: 1M
 Shared window ifc-protomax: 1M
 Shared window ifc-mpls: 7M
 Shared window ifc-ipv6: 6M
 Shared window ifc-ipv4: 10M
 Shared window mfwdv6: 449K
 Shared window mfwd_info: 733K
 Shared window infra_statsd: 3K
 Shared window im_rd: 1M
 Shared window im_db: 1M
```

```
Shared window infra_ital: 67K
Shared window rspp_ma: 3K
Shared window aib: 623K
Shared window im_rules: 293K
Shared window ees_fsdb_svr: 720K
--More--
```

Table 6-2 Heading Descriptions for show memory Command Output

Heading	Description
Physical Memory	Amount of physical memory installed on the device.
Application Memory	Memory available for the system to use (total memory minus image size, reserved, IOMem, and flashfsys).
Image	Size of the bootable image.
Reserved	Reserved for packet memory.
IOMem	IO memory—Currently used as a backup for packet memory.
flashfsys	Flash file system memory.
Process and JID	Process and job ID.
Address	Starting address in memory.
Bytes	Size of memory block.
What	Block description.

Removing Configurations to Resolve Low-Memory Warnings

To resolve most low-memory problems, you should remove the configurations from the router that are consuming the most memory. Often, memory problems occur when a large new configuration is added to the system. The following sections provide information to resolve low-memory issues:

- Clearing a Target Configuration, page 6-130
- Removing Committed Configurations to Free System Memory, page 6-131
- Rolling Back to a Previously Committed Configuration, page 6-131
- Clearing Configuration Sessions, page 6-131

Clearing a Target Configuration

A low-memory warning can occur when a large configuration file is loaded into a target configuration session. To remove the target configuration, enter the **clear** command to discard the changes. For example:

```
RP/0/0/CPU0:router(config)# clear
```

Caution Committing a target configuration that has caused a low-memory warning can make the system unstable. Clearing a target configuration is a preventive measure to not let the system go into a worse memory state due to additional configuration. In addition, all other active configuration sessions can be closed to minimize the churn.

Removing Committed Configurations to Free System Memory

You can reduce memory usage by removing configurations from the router, as shown in the following procedure:

Step 1 Enter the **show memory summary** command in EXEC mode to display the overall system memory:

```
RP/0/0/CPU0:router# show memory summary

Tue Jun  1 00:06:34.583 DST
Physical Memory: 4096M total (2020M available)
 Application Memory : 3818M (2020M available)
 Image: 50M (bootram: 50M)
 Reserved: 226M, IOMem: 2028M, flashfsys: 0
 Total shared window: 32M
```

Step 2 Enter the **show configuration commit list** command in EXEC or administration EXEC mode to list the configurations you can remove.

Note To display the details of a configuration, enter the **show configuration commit changes** command followed by a commitID number. To display additional configuration history information, enter the **show configuration history ?** command, and use the command options to display additional information.

Step 3 Enter the **show running-config** command to display the current configuration.

Step 4 Remove configurations as needed to free memory.

For more information, see Managing Configuration History and Rollback, page 4-81.

Rolling Back to a Previously Committed Configuration

You can roll back the system to a previous committed configuration, as described in Managing Configuration History and Rollback, page 4-81.

Clearing Configuration Sessions

Active configuration sessions and their associated target configurations can consume system memory. Users with the appropriate access privileges can display the open configuration sessions of other users and terminate those sessions, if necessary (see Table 6-3).

Table 6-3 Session Commands

Command	Description
show configuration sessions	Displays the active configuration sessions.
clear configuration sessions *session-id*	Clears a configuration session.

In the following example, the open configuration sessions are displayed with the **show configuration sessions** command. The **clear configuration sessions** command is then used to clear a configuration session.

```
RP/0/0/CPU0:router# show configuration sessions
```

```
Session                   Line        User      Date                      Lock
00000211-002c409b-00000000  con0_RP1_CPU0  UNKNOWN   Mon Feb  2 01:02:09 2004

RP/0/0/CPU0:router# clear configuration sessions 00000211-002c409b-00000000

session ID '00000211-002cb09b-00000000' terminated
```

Contacting TAC for Additional Assistance

If you remove configurations and the low-memory condition remains, you may need to contact TAC for additional assistance. See the "Additional Sources of Information" section on page 6-119.

Interfaces Not Coming Up

The router interfaces are directly used in processing network traffic, so their status information is crucial to understanding how the device is functioning. This section contains information on the EXEC mode commands used to verify that the router interfaces are operational. The basic commands used in this process are summarized in Table 6-4.

Table 6-4 show interface Commands

Command	Description
show interfaces	Displays detailed information about all interfaces installed or configured on the device, whether or not they are operational.
show interfaces *type instance*	Specifies a particular interface, rather than displaying information for all interfaces, as in the following example: `show interface POS0/1/0/0`
show ipv4 interface **show ipv6 interface**	Displays basic, IP-related information for all available interfaces.
show ipv4 interface brief **show ipv6 interface brief**	Quickly displays the most critical information about the interfaces, including the interface status (up or down) and the protocol status.

Verifying the System Interfaces

Perform the following steps to verify the system interfaces.

Step 1 Enter the **show platform** command in administration EXEC to verify that all nodes are in the "IOS XR RUN" state:

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


Mon May 31 23:51:18.181 DST
Node            Type            PLIM            State           Config State
--------------------------------------------------------------------------------
0/0/CPU0        PRP(Active)     N/A             IOS XR RUN      PWR,NSHUT,MON
0/1/CPU0        L3 Service Eng  N/A             Admin Down      PWR,SHUT,MON
0/2/CPU0        L3LC Eng 3      OC3-ATM-4       IOS XR RUN      PWR,NSHUT,MON
```

```
0/3/CPU0        L3LC Eng 5+     Jacket Card     IOS XR RUN      PWR,NSHUT,MON
0/3/1           SPA             SPA-IPSEC-2G-2  READY           PWR,NSHUT
0/3/2           SPA             SPA-1XCHSTM1/OC READY           PWR,NSHUT
0/4/CPU0        L3LC Eng 5      Jacket Card     IOS XR RUN      PWR,NSHUT,MON
0/4/0           SPA             SPA-5X1GE       READY           PWR,NSHUT
0/17/CPU0       CSC6(P)         N/A             PWD             PWR,NSHUT,MON
0/18/CPU0       SFC6            N/A             PWD             PWR,NSHUT,MON
0/19/CPU0       SFC6            N/A             PWD             PWR,NSHUT,MON
0/20/CPU0       SFC6            N/A             PWD             PWR,NSHUT,MON
0/24/CPU0       ALARM6          N/A             PWD             PWR,NSHUT,MON
0/25/CPU0       ALARM6          N/A             PWD             PWR,NSHUT,MON
0/28/CPU0       GSR6-BLOWER     N/A             PWD             PWR,NSHUT,MON
```

Note When the **show platform** command is entered in EXEC mode, the display shows only those nodes assigned to the SDR.

Step 2 Enter the **show ipv4 interface brief** command to verify the IP address configuration and protocol status:

```
RP/0/0/CPU0:router# show ipv4 interface brief

Interface                      IP-Address      Status                Protocol
POS0/1/0/0                     unassigned      Shutdown              Down
POS0/1/0/1                     unassigned      Shutdown              Down
POS0/1/0/2                     unassigned      Shutdown              Down
POS0/1/0/3                     unassigned      Shutdown              Down
POS0/1/0/4                     unassigned      Shutdown              Down
POS0/1/0/5                     unassigned      Shutdown              Down
POS0/1/0/6                     unassigned      Shutdown              Down
POS0/1/0/7                     unassigned      Shutdown              Down
POS0/1/0/8                     unassigned      Shutdown              Down
POS0/1/0/9                     unassigned      Shutdown              Down
POS0/1/0/10                    unassigned      Shutdown              Down
POS0/1/0/11                    unassigned      Shutdown              Down
POS0/1/0/12                    unassigned      Shutdown              Down
POS0/1/0/13                    unassigned      Shutdown              Down
POS0/1/0/14                    unassigned      Shutdown              Down
POS0/1/0/15                    unassigned      Shutdown              Down
POS0/2/0/0                     10.10.1.101     Down                  Down
POS0/2/0/1                     unassigned      Shutdown              Down
POS0/2/0/2                     unassigned      Shutdown              Down
POS0/2/0/3                     unassigned      Shutdown              Down
TenGigE0/3/0/0                 unassigned      Shutdown              Down
TenGigE0/3/0/2                 unassigned      Shutdown              Down
MgmtEth0/RP0/CPU0/0            unassigned      Shutdown              Down
```

Step 3 Configure the interfaces, as shown in the following examples.

Note You must enter the **commit** command to make the new configuration part of the active running configuration. If you end the configuration session, you are automatically prompted to commit the changes, as shown in the second example:

```
RP/0/0/CPU0:router# configure
RP/0/0/CPU0:router(config)# interface pos0/2/0/1
RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.1.1 255.0.0.0
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# commit
RP/0/0/CPU0:router(config-if)# end
RP/0/0/CPU0:router#

RP/0/0/CPU0:router# configure
```

```
RP/0/0/CPU0:router(config)# interface pos0/2/0/2
RP/0/0/CPU0:router(config-if)# ipv4 address 10.1.1.2 255.255.0.0
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# end
Uncommitted changes found, commit them? [yes]: yes
RP/0/0/CPU0:router#
```

Step 4 Enter the **show ipv4 interface brief** command to verify that the interfaces are "Up" in the Status column:

```
RP/0/0/CPU0:router# show ipv4 interface brief

Interface                      IP-Address      Status              Protocol
POS0/1/0/0                     unassigned      Shutdown            Down
POS0/1/0/1                     unassigned      Shutdown            Down
POS0/1/0/2                     unassigned      Shutdown            Down
POS0/1/0/3                     unassigned      Shutdown            Down
POS0/1/0/4                     unassigned      Shutdown            Down
POS0/1/0/5                     unassigned      Shutdown            Down
POS0/1/0/6                     unassigned      Shutdown            Down
POS0/1/0/7                     unassigned      Shutdown            Down
POS0/1/0/8                     unassigned      Shutdown            Down
POS0/1/0/9                     unassigned      Shutdown            Down
POS0/1/0/10                    unassigned      Shutdown            Down
POS0/1/0/11                    unassigned      Shutdown            Down
POS0/1/0/12                    unassigned      Shutdown            Down
POS0/1/0/13                    unassigned      Shutdown            Down
POS0/1/0/14                    unassigned      Shutdown            Down
POS0/1/0/15                    unassigned      Shutdown            Down
POS0/2/0/0                     10.10.1.101     Up                  Up
POS0/2/0/1                     10.1.1.1        Up                  Up
POS0/2/0/3                     10.1.1.2        Shutdown            Down
POS0/2/0/3                     unassigned      Shutdown            Down
TenGigE0/3/0/0                 unassigned      Shutdown            Down
TenGigE0/3/0/2                 unassigned      Shutdown            Down
MgmtEth0/RP0/CPU0/0            unassigned      Shutdown            Down
```

Step 5 If the interface is in the "Shutdown/Down" state, as shown in the previous example, perform the following tasks:

a. Verify that the status of the interface is "Shutdown":

```
RP/0/0/CPU0:router# show running-config interface POS0/2/0/3

interface pos0/2/0/3
 shutdown
 keepalive disable
!
```

b. Bring the interface up with the following commands:

```
RP/0/0/CPU0:router(config)# controller SONET 0/2/0/3
RP/0/0/CPU0:router(config-sonet)# no shutdown
RP/0/0/CPU0:router(config-sonet)# commit
RP/0/0/CPU0:router(config-sonet)# exit

RP/0/0/CPU0:router(config)# interface pos 0/2/0/3
RP/0/0/CPU0:router(config-if)# no shutdown
RP/0/0/CPU0:router(config-if)# commit
RP/0/0/CPU0:router(config-if)# end
RP/0/0/CPU0:router#
```

Step 6 If the interface state is still displayed as "Down", verify that the physical cable connections are correctly installed. The following message indicates that the interface has either a bad connection or no connection:

```
LC/0/0/1:Sep 29 15:31:12.921 : plim_4p_oc192[183]: %SONET-4-
ALARM : SONET0_1_1_0: SLOS
```

Step 7 Verify again that the interface is up by entering the **show ipv4 interface brief** command:

```
RP/0/0/CPU0:router# show ipv4 interface brief

Interface                      IP-Address      Status                Protocol
POS0/1/0/0                     unassigned      Shutdown              Down
POS0/1/0/1                     unassigned      Shutdown              Down
POS0/1/0/2                     unassigned      Shutdown              Down
POS0/1/0/3                     unassigned      Shutdown              Down
POS0/1/0/4                     unassigned      Shutdown              Down
POS0/1/0/5                     unassigned      Shutdown              Down
POS0/1/0/6                     unassigned      Shutdown              Down
POS0/1/0/7                     unassigned      Shutdown              Down
POS0/1/0/8                     unassigned      Shutdown              Down
POS0/1/0/9                     unassigned      Shutdown              Down
POS0/1/0/10                    unassigned      Shutdown              Down
POS0/1/0/11                    unassigned      Shutdown              Down
POS0/1/0/12                    unassigned      Shutdown              Down
POS0/1/0/13                    unassigned      Shutdown              Down
POS0/1/0/14                    unassigned      Shutdown              Down
POS0/1/0/15                    unassigned      Shutdown              Down
POS0/2/0/0                     10.10.1.101     Up                    Up
POS0/2/0/1                     10.1.1.1        Up                    Up
POS0/2/0/2                     10.1.1.2        Up                    Up
POS0/2/0/3                     unassigned      Shutdown              Down
TenGigE0/3/0/0                 unassigned      Shutdown              Down
TenGigE0/3/0/2                 unassigned      Shutdown              Down
MgmtEth0/RP0/CPU0/0            unassigned      Shutdown              Down
```

Step 8 Repeat these steps for every interface, until every interface shows both Status and Protocol as “Up.”

APPENDIX A

Understanding Regular Expressions, Special Characters, and Patterns

This appendix describes the regular expressions, special or wildcard characters, and patterns that can be used with filters to search through command output. The filter commands are described in the Filtering show Command Output.

Contents

Regular Expressions

A regular expression is a pattern (a phrase, number, or more complex pattern).

- Regular expressions are case sensitive and allow for complex matching requirements. Simple regular expressions include entries like Serial, misses, or 138.
- Complex regular expressions include entries like 00210... , (is), or [Oo]utput.

A regular expression can be a single-character pattern or multiple-character pattern. It can be a single character that matches the same single character in the command output or multiple characters that match the same multiple characters in the command output. The pattern in the command output is referred to as a string.

The simplest regular expression is a single character that matches the same single character in the command output. Letters (A to Z and a to z), digits (0 to 9), and other keyboard characters (such as ! or ~) can be used as a single-character pattern.

Special Characters

Certain keyboard characters have special meaning when used in regular expressions. Table A-1 lists the keyboard characters that have special meaning.

Table A-1 Characters with Special Meaning

Character	Special Meaning
.	Matches any single character, including white space.
*	Matches 0 or more sequences of the pattern.
+	Matches 1 or more sequences of the pattern.
?	Matches 0 or 1 occurrences of the pattern.
^	Matches the beginning of the string.
$	Matches the end of the string.
_ (underscore)	Matches a comma (,), left brace ({), right brace (}), left parenthesis (( ), right parenthesis ( )), the beginning of the string, the end of the string, or a space.

To use these special characters as single-character patterns, remove the special meaning by preceding each character with a double backslash (\\). In the following examples, single-character patterns matching a dollar sign, an underscore, and a plus sign, respectively, are shown.

\\$ _ \\+

Character Pattern Ranges

A range of single-character patterns can be used to match command output. To specify a range of single-character patterns, enclose the single-character patterns in square brackets ([ ]). Only one of these characters must exist in the string for pattern-matching to succeed. For example, **[aeiou]** matches any one of the five vowels of the lowercase alphabet, while **[abcdABCD]** matches any one of the first four letters of the lowercase or uppercase alphabet.

You can simplify a range of characters by entering only the endpoints of the range separated by a dash (–), as in the following example:

[a–dA–D]

To add a dash as a single-character pattern in the search range, precede it with a double backslash:

[a–dA–D\\–]

A bracket (]) can also be included as a single-character pattern in the range:

[a–dA–D\\–\\]]

Invert the matching of the range by including a caret (^) at the start of the range. The following example matches any letter except the ones listed:

[^a–dqsv]

The following example matches anything except a right square bracket (]) or the letter d:

[^\\]d]

Multiple-Character Patterns

Multiple-character regular expressions can be formed by joining letters, digits, and keyboard characters that do not have a special meaning. With multiple-character patterns, order is important. The regular expression **a4%** matches the character **a** followed by a **4** followed by a **%**. If the string does not have a4%, in that order, pattern matching fails.

The multiple-character regular expression **a.** uses the special meaning of the period character to match the letter a followed by any single character. With this example, the strings ab, a!, and a2 are all valid matches for the regular expression.

Put a backslash before the keyboard characters that have special meaning to indicate that the character should be interpreted literally. Remove the special meaning of the period character by putting a backslash in front of it. For example, when the expression **a\\.** is used in the command syntax, only the string a. is matched.

A multiple-character regular expression containing all letters, all digits, all keyboard characters, or a combination of letters, digits, and other keyboard characters is a valid regular expression. For example: **telebit 3107 v32bis**.

Complex Regular Expressions Using Multipliers

Multipliers can be used to create more complex regular expressions that instruct Cisco IOS XR software to match multiple occurrences of a specified regular expression. Table A-2 lists the special characters that specify “multiples” of a regular expression.

Table A-2 Special Characters Used as Multipliers

Character	Description
*	Matches 0 or more single-character or multiple-character patterns.
+	Matches 1 or more single-character or multiple-character patterns.
?	Matches 0 or 1 occurrences of a single-character or multiple-character pattern.

The following example matches any number of occurrences of the letter a, including none:

a*

The following pattern requires that at least one occurrence of the letter a in the string be matched:

a+

The following pattern matches the string bb or bab:

ba?b

The following string matches any number of asterisks (*):

******

To use multipliers with multiple-character patterns, enclose the pattern in parentheses. In the following example, the pattern matches any number of the multiple-character string ab:

(ab)*

As a more complex example, the following pattern matches one or more instances of alphanumeric pairs:

([A-Za-z][0-9])+

The order for matches using multipliers (*, +, and ?) is to put the longest construct first. Nested constructs are matched from outside to inside. Concatenated constructs are matched beginning at the left side of the construct. Thus, the regular expression matches A9b3, but not 9Ab3 because the letters are specified before the numbers.

Pattern Alternation

Alternation can be used to specify alternative patterns to match against a string. Separate the alternative patterns with a vertical bar (|). Only one of the alternatives can match the string. For example, the regular expression **codex|telebit** matches the string codex or the string telebit, but not both codex and telebit.

Anchor Characters

Anchoring can be used to match a regular expression pattern against the beginning or end of the string. Table A-3 shows that regular expressions can be anchored to a portion of the string using the special characters.

Table A-3 Special Characters Used for Anchoring

Character	Description
^	Matches the beginning of the string.
$	Matches the end of the string.

For example, the regular expression **^con** matches any string that starts with con, and **sole$** matches any string that ends with sole.

In addition to indicating the beginning of a string, the ^ can be used to indicate the logical function “not” when used in a bracketed range. For example, the expression **[^abcd]** indicates a range that matches any single letter, as long as it is not the letters a, b, c, and d.

Underscore Wildcard

Use the underscore to match the beginning of a string (^), the end of a string ($), parentheses (( )) , space ( ), braces ({ }), comma (,), and underscore (_). The underscore can be used to specify that a pattern exists anywhere in the string. For example, **_1300_** matches any string that has 1300 somewhere in the string and is preceded by or followed by a space, brace, comma, or underscore. Although **_1300_** matches the regular expression {1300_, it does not match the regular expressions 21300 and 13000t.

The underscore can replace long regular expression lists. For example, instead of specifying **^1300( ) ( )1300$ {1300, ,1300, {1300} ,1300, (1300**, simply specify **_1300_**.

Parentheses Used for Pattern Recall

Use parentheses with multiple-character regular expressions to multiply the occurrence of a pattern. The Cisco IOS XR software can remember a pattern for use elsewhere in the regular expression.

To create a regular expression that recalls a previous pattern, use parentheses to indicate memory of a specific pattern and a backslash (\\) followed by a digit to reuse the remembered pattern. The digit specifies the occurrence of a parenthesis in the regular expression pattern. When there is more than one remembered pattern in the regular expression, \\1 indicates the first remembered pattern, \\2 indicates the second remembered pattern, and so on.

The following regular expression uses parentheses for recall:

a(.)bc(.)\\1\\2

This regular expression matches an a followed by any character (call it character number 1), followed by bc followed by any character (character number 2), followed by character number 1 again, followed by character number 2 again. So, the regular expression can match aZbcTZT. The software remembers that character number 1 is Z and character number 2 is T, and then uses Z and T again later in the regular expression.

GLOSSARY

A

AAA authentication, authorization, and accounting. A network security service that provides the primary framework to set up access control on a Cisco CRS router or access server. AAA is an architectural framework and modular means of configuring three independent but closely related security functions in a consistent manner.

ACL access control list. A list kept by routers to control access to or from the router for a number of services (for example, to prevent packets with a certain IP address from leaving a particular interface on the router).

active Denotes a card or process that performs a system task; in a redundant configuration, there is an inactive standby card or process available to become active. Active cards or processes are also sometimes denoted as primary.

active RP The RP that is active in a redundant pair of RPs.

active software configuration The software configuration marked as active for a node.

active software set The set of Cisco IOS XR software packages activated in one or more nodes in a router.

algorithm A well-defined rule or process for arriving at a solution to a problem. In networking, algorithms commonly are used to determine the best route for traffic from a particular source to a particular destination.

APS automatic protection switching. A method that allows transmission equipment to recover automatically from failures, such as a cut cable.

ASIC application-specific integrated circuit. A chip designed for use in a specific hardware device. An ASIC is a chip designed for a special application, such as a particular kind of transmission protocol.

B

bandwidth The amount of data that can be sent in a fixed amount of time. For digital devices, the bandwidth is usually expressed in bits per second (Bps) or bytes per second.

BGP Border Gateway Protocol. A routing protocol used between autonomous systems. It is the routing protocol that makes the internet work. BGP is a distance-vector routing protocol that carries connectivity information and an additional set of BGP attributes. These attributes allow for a rich set of policies for deciding the best route to use to reach a given destination.

C

card type The type of the card inserted in a slot.

CDP Cisco Discovery Protocol. CDP runs on all Cisco devices so that these devices can learn about neighboring devices and exchange information. CDP uses a well-known multicast MAC address. During system initialization, the application-specific integrated circuit (ASIC) is configured to forward these packets to the Cisco IOS XR software CPU, which processes the packets.

Cisco.com The Cisco website

CLI command-line interface. A text-based user interface to an operating system. A command-line interface is a user interface to a computer operating system or an application in which the user responds to a visual prompt by typing a command on a specified line, receives a response from the system, and then enters another command, and so forth. Typically, most of the UNIX-based systems today offer both a command-line interface and graphical user interface (GUI). See also *GUI*.

committed/saved software configuration The configuration stored in the system for a particular node. The RP loads the committed configuration into memory at startup.

configuration register In Cisco routers, a 16-bit, user-configurable value that determines how the router functions during initialization. The configuration register can be stored in hardware or software. In hardware, the bit position is set using a jumper. In software, the bit position is set by specifying a hexadecimal value using configuration commands. A hexadecimal or decimal value that represents the 16-bit configuration register value that you want to use the next time the router is restarted. The value range is from 0x0 to 0xFFFF (0 to 65535 in decimal).

control plane The control plane oversees the operation of the data plane, allocating resources, providing information, and handling errors to allow data plane operations to be continuous and efficient.

CORBA Common Object Request Broker Architecture. Specification that provides the standard interface definition between OMG-compliant objects. CORBA allows applications to communicate with one another no matter where they are located or who has designed them.

CoS class of service. An indication of how an upper-layer protocol requires a lower-layer protocol to treat its messages. In SNA subarea routing, CoS definitions are used by subarea nodes to determine the optimal route to establish a given session. A CoS definition comprises a virtual route number and transmission priority field. Repetitive, regularly timed signals are used to control synchronous processes.

D

DDTS distributed defect tracking system. A method to track software errors and resolutions.

DHCP Dynamic Host Configuration Protocol. Provides a mechanism for allocating IP addresses dynamically so that addresses can be reused when hosts no longer need them.

DIMM dual in-line memory module. Small circuit boards carrying memory integrated circuits, with signal and power pins on both sides of the board, in contrast to single-in-line memory modules (SIMMs).

disk0 Name of the flash disk on which the Cisco IOS XR software is stored.

disk1 Name of the optional flash disk on which the Cisco IOS XR software can be stored in preparation for installation or upgrade.

DNS Domain Name System. Mechanism used in the Internet and on private intranets for translating names of host computers into addresses. The DNS also allows host computers not directly on the Internet to have a registered name in the same style.

DPT Dynamic Packet Transport. DPT rings are dual, counter-rotating fiber rings. Both fibers are used concurrently to transport both data and control traffic.

DSC designated shelf controller. The RP that controls a standalone router or a multishelf system. The DSC is selected from among the route processors (RPs) installed in the router or multishelf system.

E

eBGP external Border Gateway Protocol. BGP sessions are established between routers in different autonomous systems. eBGPs communicate among different network domains.

ECC error correction code. ECC is used to correct errors within memories on the Cisco CRS router.

egress Outgoing channel.

Ethernet Baseband LAN specification invented by Xerox Corporation and developed jointly by Xerox, Intel, and Digital Equipment Corporation. Ethernet networks use CSMA/CD and run over a variety of cable types at 10 Mbps. Ethernet standards are defined by the IEEE 802.3 specification.

F

fabric Connectivity between all line cards. Also referred to as switch fabric.

FC fan controller. Two fan controller cards are installed in every line card chassis as a redundant pair to manage the fan assemblies; a BITS timing connector exists on the fan controller card.

FIB Forwarding Information Base. Database that stores information about switching of data packets. A FIB is based on information in the Routing Information Base (RIB). It is the optimal set of selected routes that are installed in the line cards for forwarding. See also *RIB*.

flooding Traffic-passing technique used by switches and bridges in which traffic received on an interface is sent out all the interfaces of that device except the interface on which the information was originally received.

forwarding Process of sending a frame toward its ultimate destination by way of an internetworking device.

FRR fast reroute. Automatically reroutes traffic on a label switched path (LSP) if a node or link in an LSP fails. FRR reduces the loss of packets traveling over an LSP.

FTP File Transfer Protocol. Application protocol, part of the TCP/IP protocol stack, used for transferring files between network nodes. FTP is defined in RFC 959.

G

GE Gigabit Ethernet. Standard for a high-speed Ethernet, approved by the IEEE 802.3z standards committee in 1996.

Gigabit Ethernet Standard for a high-speed Ethernet, approved by the IEEE 802.3z standards committee in 1996.

GUI graphical user interface. A user environment that uses pictorial and textual representations of the input and output of applications and the hierarchical or other data structure in which information is stored. Such conventions as buttons, icons, and windows are typical, and many actions are performed using a pointing device (such as a mouse). Microsoft Windows and the Apple Macintosh are prominent examples of platforms using a GUI. See also *CLI*.

H

HA High availability is defined as the continuous operation of systems. For a system to be available, all components, including application and database servers, storage devices, and the end-to-end network, need to provide continuous service.

HDLC high-level data link control. ISO communications protocol used in X.25 packet-switching networks. HDLC provides error correction at the data link layer and contains the following subsets: LAPB and SDLC.

hexadecimal A number system having 16 as its base. This number representation uses the digits 0–9, with their usual meaning, plus the letters A–F (or a–f) to represent hexadecimal digits with values of (decimal) 10 to 15. The far right digit counts ones, the next counts multiples of 16, then 16^2 = 256, and so on.

Hexadecimal is more succinct than binary for representing bit masks, machines addresses, and other low-level constants but it is still reasonably easy to split a hex number into different bit positions. For example, the top 16 bits of a 32-bit word are the first four hex digits.

hop Passage of a data packet between two network nodes (for example, between two routers). See also *hop count*.

hop count Routing metric used to measure the distance between a source and a destination.

HTTP Hypertext Transfer Protocol. Used by web browsers and web servers to transfer files, such as text and graphic files. HTTP is the set of rules for exchanging files (text, graphic images, sound, video, and other multimedia files) on the World Wide Web. Relative to the TCP/IP suite of protocols (which are the basis for information exchange on the Internet), HTTP is an application protocol.

I

ICMP Internet Control Message Protocol. Network layer Internet (TCP/IP) protocol that reports errors and provides other information relevant to IP packet processing.

IEP IP explicit path. List of IP addresses, each representing a node or link in the explicit path.

IETF Internet Engineering Task Force. Task force consisting of over 80 working groups responsible for developing Internet standards. The IETF operates under the auspices of ISOC.

IGMP Internet Group Management Protocol. Governs the management of multicast groups in a TCP/IP network. Used by IP hosts to report their multicast group memberships to an adjacent multicast router.

IGP Interior Gateway Protocol. Internet protocol used to exchange routing information within an autonomous system. Examples of common Internet IGPs include IGRP, OSPF, and RIP. See also *OSPF* and *RIP*.

ingress Incoming channel.

installed software set The set of Cisco IOS XR software packages installed on a router.

IOS XR The Cisco operating system used on the Cisco CRS router and Cisco XR 12000 Series Router.

IP Internet Protocol. Network layer protocol in the TCP/IP stack offering a connectionless internetwork service. IP provides features for addressing, type-of-service specification, fragmentation and reassembly, and security.

IPv4 IP Version 4. Network layer for the TCP/IP protocol suite. A connectionless, best-effort packet switching protocol.

IPv6 IP Version 6. Replacement for IPv4. A next-generation IP protocol. IPv6 is backward compatible with and designed to fix the shortcomings of IPv4, such as data security and maximum number of user addresses. IPv6 increases the address space from 32 to 128 bits, providing for an unlimited number of networks and systems. It also supports quality of service (QoS) parameters for real-time audio and video.

IPX Internetwork Packet Exchange. NetWare network layer (Layer 3) protocol used for transferring data from servers to workstations. IPX is similar to IP and XNS.

IS-IS Intermediate System-to-Intermediate System. OSI link-state hierarchical routing protocol based on DECnet Phase V routing, whereby ISs (routers) exchange routing information based on a single metric to determine network topology.

K

keepalive interval Period of time between each keepalive message sent by a network device.

keepalive message Message sent by one network device to inform another network device that the virtual circuit between the two is still active.

L

Layer 2 Layer 2 refers to the data link layer of the commonly referenced multilayered communication model, Open Systems Interconnection (OSI). The data link layer contains the address inspected by a bridge or switch. Layer 2 processing is faster than layer 3 processing, because less analysis of the packet is required.

Layer 3 Layer 3 refers to the network layer of the commonly referenced multilayered communication model, Open Systems Interconnection (OSI). The network layer is concerned with knowing the address of the neighboring nodes in the network, selecting routes and quality of service, and recognizing and forwarding to the transport layer incoming messages for local host domains.

A router is a Layer 3 device, although some newer switches also perform Layer 3 functions. The Internet Protocol (IP) address is a Layer 3 address.

LC line card.

LDP label distribution protocol. A standard protocol between MPLS-enabled routers to negotiate the labels (addresses) used to forward packets. The Cisco proprietary version of this protocol is the Tag Distribution Protocol (TDP).

LIB Label Information Base. The table that contains the labels in use on the node.

loopback Send the outgoing signals back to the receiving side for testing.

M

MAC address Standardized data link layer address that is required for every port or device that connects to a LAN. Other devices in the network use these addresses to locate specific ports in the network and to create and update routing tables and data structures. MAC addresses are 6 bytes long and are controlled by the IEEE. Also known as a hardware address, MAC layer address, and physical address.

mask Pattern of bits used to reject or accept bit patterns in another set of data.

MBI minimum boot image. Software image containing a kernel and minimum set of drivers and components to boot a node.

Mbps megabits per second. A bit rate expressed in millions of binary bits per second. 1 megabit = 2^{20} bits, or 1,048,576 bits.

MIB Management Information Base. Database of network management information that is used and maintained by a network management protocol like Simple Network Management Protocol (SNMP). The value of an MIB object can be changed or retrieved using SNMP commands, usually through a GUI network management system. MIB objects are organized in a tree structure that includes public (standard) and private (proprietary) branches.

MPLS Multiprotocol Label Switching. Switching method that forwards IP traffic using a label. This label instructs the routers and switches in the network where to forward the packets based on pre-established IP routing information

MPLS TE Multiprotocol Label Switching traffic engineering. A switching method that forwards IP traffic using a label. This label instructs the routers and switches in the network where to forward the packets based on pre-established IP routing information.

MSC modular services card. Module in which the ingress and egress packet processing and queueing functions are carried out in the Cisco CRS architecture. Up to 16 MSCs are installed in a line card chassis; each MSC must have an associated physical line interface module (PLIM) (of which there are several types to provide a variety of physical interfaces). The MSC and PLIM mate together on the line card chassis midplane. See also *PLIM*.

MSCs are also referred to as line cards.

MTU maximum transmission unit. Maximum packet size, in bytes, that a particular interface can handle.

multicast Multicast is a feature that refers to single packets copied by the network and sent to a specific subset of network addresses. These addresses are specified in the Destination Address Field. See also *unicast*.

N

netboot Loading software images from a network server, such as TFTP.

node A card installed and running on the router.

NSF nonstop forwarding. Packets keep flowing during events such as switchover, process restarts, and the upgrade or downgrade of software packages. Nonstop forwarding is the ability of a router to continue to forward traffic toward a router that may be recovering from a transient failure and the ability of a router recovering from a transient failure in the control plane to continue correctly forwarding traffic sent to it by a peer.

NTP Network Time Protocol. Protocol built on top of TCP that ensures accurate local time-keeping with reference to radio and atomic clocks located on the Internet. This protocol is capable of synchronizing distributed clocks within milliseconds over long time periods.

NVRAM nonvolatile RAM. Static random access memory that is made into nonvolatile storage by having a battery permanently connected.

O

OC-x Optical carrier, where x=3, 12, 48, or 192, relating to the various speeds within a SONET network.

OIR online insertion and removal. Feature that permits the addition, replacement, or removal of cards without interrupting the system power, entering console commands, or causing other software or interfaces to shut down. Sometimes called hot-swapping or power-on servicing.

OSI Open Systems Interconnection. International standardization program created by ISO and ITU-T to develop standards for data networking that facilitate multivendor equipment interoperability.

OSPF Open Shortest Path First. Link-state, hierarchical Interior Gateway Protocol (IGP) routing algorithm proposed as a successor to Routing Information Protocol (RIP) in the Internet community. OSPF features include least-cost routing, multipath routing, and load balancing. OSPF was derived from an early version of the Intermediate System-to-Intermediate System (IS-IS) protocol. See also *IGP* and *RIP*.

P

package A group of software components installed on the router.

packet Logical grouping of information that includes a header containing control information and (usually) user data. Packets most often are used to refer to network layer units of data.

Packet over SONET/SDH POS. Packet over SONET/SDH enables core routers to send native IP packets directly over SONET or SDH frames.

PAP Password Authentication Protocol. Authentication protocol that allows PPP peers to authenticate one another. The remote router attempting to connect to the local router is required to send an authentication request. Unlike Challenge Handshake Authentication Protocol (CHAP), PAP passes the password and the hostname or username in the clear (unencrypted). PAP does not itself prevent unauthorized access but merely identifies the remote end. The router or access server then determines whether that user is allowed access. PAP is supported only on PPP lines. See also *PPP*.

PCMCIA Personal Computer Memory Card International Association. Standard for credit card-size memory or I/O device.

PIE package installation envelope. An installable software file with the suffix *pie*. A PIE may be a package or a Software Maintenance Upgrade (SMU). A PIE is used to deliver Cisco IOS XR software. A PIE may contain a single component, group of components (called a package), or set of packages. When a PIE contains more than one component, it is called a "Composite PIE."

PLIM Physical layer interface module. Provides the physical interface for a line card. Also handles media-specific functions, such as framing, clock recovery, channelization, and optical signaling for line interfaces connecting to a Cisco CRS router.

PM performance monitoring. Provides a variety of automatic functions to aid in the maintenance and operation of the network. PM is continuous, in-service monitoring of transmission quality that uses software-provisionable performance parameters. Performance parameters are measured for all four layers of the SONET signal: physical, section, line, and STS path.

POS Packet over SONET/SDH. POS enables core routers to send native IP packets directly over Synchronous Optical Network (SONET) or Synchronous Digital Hierarchy (SDH) frames.

PPP Point-to-Point Protocol. Successor to SLIP that provides router-to-router and host-to-network connections over synchronous and asynchronous circuits. Whereas SLIP was designed to work with IP, PPP was designed to work with several network layer protocols, such as IP, IPX, and ARA. PPP also has built-in security mechanisms, such as CHAP and PAP. PPP relies on two protocols: LCP and NCP.

primary RP The first route processor configured for DSC or logical router operation. If a second RP is configured as a redundant RP, it becomes the secondary RP.

Q

QoS quality of service. A set of parameters that describes a flow of data, such as guaranteed bandwidth, delay, and delivery guarantee.

R

RCP remote copy protocol. A protocol that allows users to copy files to and from a file system residing on a remote host or server on the network. The RCP protocol uses TCP to ensure the reliable delivery of data.

RIB Routing Information Base. This is the set of all available routes from which to choose the FIB. The RIB essentially contains all routes available for selection. Essentially, it is the sum of all routes learned by dynamic routing protocols, all directly attached networks (that is. networks to which a given router has interfaces connected), and any additional configured routes, such as static routes.

RIP Routing Information Protocol. A simple routing protocol that is part of the TCP/IP protocol suite and the most common IGP in the Internet. RIP determines a route based on the smallest hop count between source and destination. It is a distance vector protocol that broadcasts routing information to neighboring routers. It is known to use excessive bandwidth. See also *hop count* and *IGP*.

ROM Monitor ROM Monitor is a bootstrap program that initializes the hardware and boots the system when a router is powered on or reset. ROM Monitor mode is also known as "ROMMON," which reflects the CLI prompts for the mode.

```
rommon B1> (Cisco CRS routers)
```

or

```
rommon1> (Cisco XR 12000 Series Routers)
```

ROMMON See ROM Monitor.

router Network layer device that uses one or more routing metrics to determine the optimal path along which network traffic should be forwarded. Routers forward packets from one network to another based on network layer information.

routing Process of finding a path to a destination host. Routing is very complex in large networks because of the many potential intermediate destinations a packet might traverse before reaching its destination host.

routing metric A routing algorithm determines that one route is better than another. This information is stored in routing tables. Metrics include bandwidth, communication cost, delay, hop count, load, MTU, path cost, and reliability. Sometimes referred to simply as a metric. See also *algorithm*.

routing protocol Protocol that accomplishes routing through the implementation of a specific routing algorithm. Examples of routing protocols include BGP, OSPF, and IS-IS.

routing table Table stored in a router or some other internetworking device that keeps track of routes to particular network destinations and, in some cases, metrics associated with those routes.

RP route processor. Cards that contain run-control software on the router. Two RPs are installed as a redundant pair in dedicated slots in the front of each line card chassis.

RPF Reverse Path Forwarding. Multicasting technique in which a multicast datagram is forwarded from all but the receiving interface if the receiving interface is the one used to forward unicast datagrams to the source of the multicast datagram.

RSVP Resource Reservation Protocol. Protocol that supports the reservation of resources across an IP network. Applications running on IP end systems can use RSVP to indicate to other nodes the nature (bandwidth, jitter, maximum burst, and so on) of the packet streams they want to receive. RSVP depends on IPv6. Also known as Resource Reservation Setup Protocol. See also *IPv6*.

running configuration The router configuration currently in effect. Although the user can save multiple versions of the router configuration for future reference, only one copy of the running configuration exists in the router at any given time.

Rx The receiver end of a fabric link. All links are unidirectional. See also *Tx*.

S

SCFC shelf controller/fan controller. Combines shelf controller function and fan controller function on one card. Two are installed in each fabric chassis.

SCGE shelf controller Gigabit Ethernet card. Gigabit Ethernet switch on a system controller card in the fabric chassis.

SDH Synchronous Digital Hierarchy. European standard that defines a set of rate and format standards that are sent using optical signals over fiber. SDH is similar to SONET, with a basic SDH rate of 155.52 Mbps, designated at STM-1.

SDR secure domain router. A collection of line cards and route processors that form a complete router. Each router contains its own instance of dynamic routing, IP stack, system database, interface manager, and event notification system.

SDRAM synchronous dynamic random access memory. A form of dynamic RAM that adds a separate clock signal to the control signals.

shelf controller The hardware component that manages the configuration and health of a fabric chassis within the Cisco CRS router.

shelf manager The shelf manager process runs on a router or switch, doing platform-dependent functions, including handling OIR events. Shelf manager is formerly called platform manager.

SMU Software Maintenance Upgrade. A “point fix” for a critical problem. SMUs are delivered as PIE files and are used to update software packages.

SNMP Simple Network Management Protocol. SNMP is the protocol governing network management and the monitoring of network devices and their functions. It is not necessarily limited to TCP/IP networks.

SNMPv3 Simple Network Management Protocol Version 3. An interoperable standards-based protocol for network management. SNMPv3 provides secure access to devices by a combination of authenticating and encrypting packets over the network.

software configuration A list of packages activated for a particular node. A software configuration consists of a boot package and additional feature packages.

SONET Synchronous Optical Network. A standard format for transporting a wide range of digital telecommunications services over optical fiber. SONET is characterized by standard line rates, optical interfaces, and signal formats. See also *SDH*.

SP service processor. An SP on each card maintains an internal management connection to the shelf controller for the rack. The SP is referred to in CLI commands to identify the nodeID for fabric, alarm and fan controller cards.

Example:
```
RP/0/RPO/CPU:router# admin show controllers fabric connectivity location 0/SM0/SP
```

SPE Synchronous Payload Envelope. Portion of the SONET frame containing overhead information (POH and user data).

SPF shortest path first. Routing algorithm that iterates on length of path to determine a shortest-path spanning tree. Commonly used in link-state routing algorithms. Sometimes called Dijkstra's algorithm.

SSH Secure Shell. A protocol that provides a secure remote connection to a router through a Transmission Control Protocol (TCP) application.

SSL secure socket layer. A secure socket between two entities with authentication.

standby Denotes an inactive card or process that waits to become active; standby cards or processes are also sometimes denoted as backup.

startup configuration The router configuration designated to be applied on the next router startup.

subinterface Virtual interfaces created on a hardware interface. These software-defined interfaces allow for segregation of traffic into separate logical channels on a single hardware interface and better utilization of the available bandwidth on the physical interface.

switchover A switch between the active and standby cards. The switchover can be initiated by command, or it can occur automatically when the active card fails.

system reload Reload of a router node.

system restart Soft reset of a router node. This involves restarting all processes running on that node.

T

TAC Cisco Technical Assistance Center.

TACACS Terminal Access Controller Access Control System. Authentication protocol, developed by the DDN community, that provides remote access authentication and related services, such as event logging. User passwords are administered in a central database rather than in individual routers, providing an easily scalable network security solution.

tar A tar file is a file produced by the UNIX tar program, which packages multiple files in a single file for distribution as a single unit. Each tar file has a *tar* filename extension.

target configuration A “two-stage” configuration of the Cisco IOS XR software running configuration. This allows users to make changes to the running configuration and accept these changes by entering the commit command.

task ID An identifier that determines user access to a given command or series of commands. A user must be a member of a group with the appropriate task IDs assigned to it to execute the related commands.

Tbps Terabits per second. The amount of data that can be sent in a fixed amount of time. 1 terabit = 2^{40} bits, or 1,099,511,627,776 bits.

TCP Transmission Control Protocol. Connection-oriented transport layer protocol that provides reliable full-duplex data transmission. TCP is part of the TCP/IP protocol stack.

Telnet Standard terminal emulation protocol in the TCP/IP protocol stack. Telnet is used for remote terminal connection, enabling users to log in to remote systems and use resources as if they were connected to a local system. Telnet is defined in RFC 854.

terabyte A unit of computer memory or data storage capacity equal to 1024 gigabytes (2^{40} bytes). Approximately 1 trillion bytes.

TFTP Trivial File Transfer Protocol. A simplified version of FTP that 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).

Note: some TFTP servers (such as Sun Solaris) may not support file sizes larger that 32 MB.

trap Message sent by an SNMP agent to an NMS, a console, or a terminal to indicate the occurrence of a significant event, such as a specifically defined condition or a threshold that was reached.

tunnel Secure communication path between two peers, such as two routers.

Tx The transmitter end of a fabric link. All links are unidirectional. See also Rx.

U

UDP User Datagram Protocol. Connectionless transport layer protocol in the TCP/IP protocol stack. UDP is a simple protocol that exchanges datagrams without acknowledgments or guaranteed delivery, requiring that error processing and retransmission be handled by other protocols. UDP is defined in RFC 768.

unicast Message sent to a single network destination.

unicast transmission A unicast transmission sends one copy of each packet to each member of the group. This method is inefficient because the same information must be carried multiple times, requiring extra bandwidth.

V

VCSEL vertical cavity surface emitting laser.

vm A vm file is a Cisco IOS XR software file that can be installed from ROM Monitor mode. A vm file is typically used to install the Cisco IOS XR software when the software has not yet been installed or has been corrupted.

VPN Virtual Private Network. Enables IP traffic to travel securely over a public TCP/IP network by encrypting all traffic from one network to another. A VPN uses “tunneling” to encrypt all information at the IP level.

W

WRED Weighted Random Early Detection. Queueing method that ensures that high-precedence traffic has lower loss rates than other traffic during times of congestion.

X

XML Extensible Markup Language. A standard maintained by the World Wide Web Consortium (W3C) that defines a syntax that lets you create markup languages to specify information structures. Information structures define the type of information, for example, subscriber name or address, not how the information looks (bold, italic, and so on). External processes can manipulate these information structures and publish them in a variety of formats. XML allows you to define your own customized markup language.

XML agent A process on the router that is sent XML requests by XML clients and is responsible for carrying out the actions contained in the request and returning an XML response back to the client.

XML client An external application that sends an XML request to the router and receives XML responses to those requests.

XML operation A portion of an XML request that specifies an operation that the XML client would like the XML agent to perform.

XML operation provider The router code that carries out a particular XML operation including parsing the operation XML, performing the operation, and assembling the operation XML response

XML request An XML document sent to the router containing a number of requested operations to be carried out.

XML response The response to an XML request.

XML schema An XML document specifying the structure and possible contents of XML elements that can be contained in an XML document.

INDEX

Symbols

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

W

X