Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide

January 2008

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-5070-08

CONTENTS

Preface

This preface describes the objectives and organization of this document and explains how to find additional information on related products and services. This preface contains the following sections:

Objectives

This document describes the configuration and troubleshooting of SPA interface processors (SIPs), SPA services cards (SSCs), and shared port adapters (SPAs) that are supported on a Cisco 7600 series router.

Document Revision History

The Document Revision History records technical changes to this document. The table shows the Cisco IOS software release number and document revision number for the change, the date of the change, and a brief summary of the change.

Release No.	Revision	Date	Change Summary
12.2(33)SRC			Support was added for the following features: • CT3 CEoP on c7600-SIP-400 • Accelerated Lawful Intercept on Cisco 7600 SIP-400 • CoPP Enhancements of Cisco 7600 SIP-400 • PPPoEoE on Cisco 7600 SIP-400 • Source IPv4 and Source MAC Address Binding on Cisco 7600 SIP-400 • IMA on SIP-400 for 24xT1/E1 CEOP and 1xOC3 CEOP SPAs • IGMP Snooping support on SIP-200 • AFC and PFC support on Multilink Interface on SIP-200 for 2- and 4-port CT3, 8-port channelized T1/E1 channelized, and 1-port channelized OC3/STM-1 SPAs • Programmable BERT patterns enhancement on SIP-200 for 2- and 4-port channelized T3 and 1-port channelized OC3/STM-1 SPAs • TDM Local switching • Phase 2 Access Circuit Redundancy with Local Switching • SPA-1xCHSTM1/OC3 • Cisco Channelized T3 to DS0 Shared Port Adapter (SPA-2XCT3/DS0, SPA-4XCT3/DS0) • Cisco 8-Port Channelized T1/E1 Shared Port Adapter (SPA-8XCHT1/E1) • Cisco Clear Channel T3/E3 Shared Port Adapter (SPA-2XT3/E3, SPA-4XT3/E3)

12.2(33)SRB1	OL-5070-07	June 4, 2007	Support for the following features was introduced: • Backup interface for Flexible UNI (for Gigabit Ethernet SPAs) on a Cisco 7600 SIP-400 • Any Transport over MPLS over GRE (AToM over GRE) on a Cisco 7600 SIP-400 • MTU support on MLPPP interfaces on a Cisco 7600 SIP-200 • ATM pseudowire redundancy for the CEoP SPA • Out-of-band clocking for the CEoP SPA • Support for XFP-10GZR-OC192LR
12.2(33)SRB	OL-5070-06	February 27, 2007	Sixth release. Support for the following features was introduced: • Software-based MLP bundles from 256 to 1024 on a Cisco 7600 SIP-200 • Network clock support on a Cisco 7600 SIP-200 • Lawful Intercept on a Cisco 7600 SIP-400 • Per-subscriber/per-protocol CoPP support on a Cisco 7600 SIP-400 • Security ACLs on a Cisco 7600 SIP-400 • Percent priority/percent bandwidth support on a Cisco 7600 SIP-400 • IGMP/PIM snooping for VPLS pseudowire on a Cisco 7600 SIP-400 • Dual-priority queue support on a Cisco 7600 SIP-400 • 24-Port Channelized T1/E1 ATM CEoP SPA, 1-Port Channelized OC-3/STM1 ATM CEoP SPAs, and 2-Port Copper and Optical Gigabit Ethernet SPAs.

<table>
<tr>
<td>12.2(33)SRA</td>
<td>OL-5070-05</td>
<td>June 5, 2006</td>
<td>Fifth release. The following modifications were made:

Support was added for the following SPAs on the Cisco 7600 SIP-200:

1-Port Channelized OC-3/STM-1 SPA
4-Port and 8-Port Fast Ethernet SPA

Support was added for the 1-Port OC-48c/STM-16 POS SPA on the Cisco 7600 SIP-400
Support was added for the 2-Port and 4-Port OC-48c/STM-16 POS SPA on the Cisco 7600 SIP-600
The following features were introduced for the IPSec VPN SPA:

Front-side VRF
IPSec Virtual Tunnel Interface (VTI)
Certificate to ISAKMP Profile Mapping
Call Admission Control
Periodic Message Option (now supported in Dead Peer Detection)
Reverse Route Injection (RRI)
IPSec Anti-replay Windowsize
IPSec Preferred Peer
Local Certificate Storage Location
Optional OCSP Nonces
Persistent Self-signed Certificates
Certificate Chain Verification
Easy VPN Remote RSA Signature Storage
IPSec and IKE MIB support for Cisco VRF-Aware IPSec

Note Support is not included for IPSec stateful failover using HSRP and SSP.</td>
</tr>
</table>

12.2(33)SRA	OL-5070-05	June 5, 2006	• The single configuration chapter for the IPSec VPN SPA has been restructured into seven smaller chapters. • Support for the following features was introduced on the Cisco 7600 SIP-200: – AToM VP Mode Cell Relay—ATM SPAs – BCP over dMLPPP (Trunk Mode)—Channelized SPAs – MPLS over RBE—ATM SPAs – Multi-VC to VLAN scalability – QoS support on bridging features – Software-based MLPPP – Software-based MLFR • Support for the following features was introduced on the Cisco 7600 SIP-400: – AToM VP Mode Cell Relay—ATM SPAs – Ethernet over MPLS (EoMPLS) VC Scaling—Increase from 4K to 10K VCs – Ingress/Egress CoS classification with ingress policing per VLAN or EoMPLS VC – Hierarchical VPLS (H-VPLS) with MPLS Edge – Hierarchical QoS support for EoMPLS VCs – Multipoint Bridging (MPB) for Gigabit Ethernet SPA – Multi-VC to VLAN scalability – Multi-VLAN to VC—ATM SPAs – QoS support on bridging features – Tag-Native Mode for Trunk BCP

12.2(18)SXF2	OL-5070-04	February 28, 2006	The following updates were made to the documentation: • Removed the restriction of "Mapping DSCP values to MPLS EXP bits is not supported" from the Cisco 7600 SIP-600 list of restrictions. • Added the following VPLS scalability support information for the Cisco 7600 SIP-600: – Up to 4000 VPLS domains – Up to 60 VPLS peers per domain – Up to 30,000 pseudowires, used in any combination of domains and peers up to the 4000-domain or 60-peer maximums. For example, support of up to 4000 domains with 7 peers or up to 60 peers in 500 domains. • Added H-VPLS with Q-in-Q edge feature support on Cisco 7600 SIP-600—Requires Cisco 7600 SIP-600 in the uplink, and any LAN port or Cisco 7600 SIP-600 on the downlink • Removed VPLS pseudowire redundancy feature support for the Cisco 7600 SIP-600 • Removed the "Cisco 7600 SIP-600 MPLS Marking" section • Modified the encapsulations supported in the ATM chapters to "aal5snap" only • Corrected the note in the "Configuring Compressed Real-Time Protocol" section of Chapter 4, "Configuring the SIPs and SSC" to state: "cRTP is supported only on the Cisco 7600 SIP-200 with the 8-Port Channelized T1/E1 SPA and 2-Port and 4-Port Channelized T3 SPA."
12.2(18)SXF2	OL-5070-04	January 27, 2006	The following update to the hardware-based MLPPP LFI guidelines was made in Chapter 17, "Configuring the 8-Port Channelized T1/E1 SPA," and Chapter 19, "Configuring the 2-Port and 4-Port Channelized T3 SPAs": • When hardware-based LFI is enabled, fragmentation counters are not displayed.

12.2(18)SXF2	OL-5070-04	January 20, 2006	Fourth release. The following modifications were made: • The 1-Port OC-192c/STM-64 POS/RPR VSR Optics SPA was introduced on the Cisco 7600 SIP-600. • Support was introduced for the configuration of IP multicast over a GRE tunnel on the IPSec VPN SPA. • Support for the "Enhancements to RFC 1483 Spanning Tree Interoperability" feature was added for ATM SPAs on the Cisco 7600 SIP-200. • Documentation of a workaround for ATM SPA configuration on the Cisco 7600 SIP-200 was added in Chapter 7, "Configuring the ATM SPAs" to address a Routed Bridge Encapsulation (RBE) limitation where only one remote MAC address is supported.
12.2(18)SXF	OL-5070-03	January 12, 2006	The following modifications were made: • Adjusted ATM SPA PVC restriction (correctly noted elsewhere in the documentation) from "A maximum number of 400 PVCs or SVCs..." to "A maximum number of 1000 PVCs or 400 SVCs configured with MQC policy maps." • Added cross-references throughout Chapter 3, "Overview of the SIPs and SSC" to the Cisco IOS Release SX Supervisor Engine release notes. • Updated the Cisco 7600 SIP-400 restrictions to clarify that the SIP does not work with the Supervisor Engine PFC3A or in PFC3A mode. • Updated the Cisco 7600 SIP-600 restrictions to clarify lack of support for the Supervisor Engine 720 PFC3A or PFC3A mode: "The Cisco 7600 SIP-600 is not supported by the Supervisor Engine 32. The Cisco 7600 SIP-600 is supported by the Supervisor Engine 720 PFC3B and Supervisor Engine 720 PFC3BXL. It is not supported with a Supervisor Engine 720 PFC3A or in PFC3A mode."

12.2(18)SXF	OL-5070-03	January 12, 2006	• Added a cross-reference to Chapter 3, "Overview of the SIPs and SSC" in each of the SPA overview chapters to ease location of additional features and restrictions that are SIP- or SSC-specific. • Removed the list of supported modules from Chapter 22, "Overview of the IPSec VPN SPA." Any unsupported modules will be documented in the "Restrictions" section. • Further qualified Cisco 7600 SIP-200 Any Transport over MPLS (AToM) support for ATM in Chapter 3, "Overview of the SIPs and SSC" to state: "Any Transport over MPLS (AToM) support, including: – ATM over MPLS (ATMoMPLS)—AAL5 VC mode – Ethernet over MPLS (EoMPLS)—(Single cell relay) VC mode" • Removed references to "1-Port 10-Gigabit Ethernet SPA and 10-Port Gigabit Ethernet SPA on a SIP-400" in the "Enabling Autonegotiation" and "Disabling Autonegotiation" sections of Chapter 12, "Configuring the Fast Ethernet and Gigabit Ethernet SPAs." • Qualified AToM core-facing restriction for the Cisco 7600 SIP-200 as follows: – AToM (ATMoMPLS, FRoMPLS, HDLCoMPLS, and PPPoMPLs) on a SPA requires a Cisco 7600 SIP-200, FlexWAN, Enhanced FlexWAN, or OSM PXF interface as the core-facing interface. – AToM (ATMoMPLS, FRoMPLS) on a Cisco 7600 SIP-200 also is supported with a Cisco 7600 SIP-400 as the core-facing interface. • Documentation of the Fast Software Upgrade (FSU) procedure supported by Route Processor Redundancy (RPR) for supervisor engines was added to Chapter 31, "Upgrading Field-Programmable Devices."

12.2(18)SXF	OL-5070-03	September 19, 2005	Third release. The following hardware was introduced: • 1-Port OC-48c/STM-16 ATM SPA • 2-Port Gigabit Ethernet SPA • 5-Port Gigabit Ethernet SPA • 10-Port Gigabit Ethernet SPA • 1-Port 10-Gigabit Ethernet SPA • 1-Port OC-192c/STM-64 POS/RPR SPA • 1-Port OC-192c/STM-64 POS/RPR XFP SPA For specific feature changes, see the Release History tables in the "Overview" chapters of this book.
12.2(18)SXE2	OL-5070-02	August 17, 2005	The following modifications were made: • Chapter 17, "Configuring the 8-Port Channelized T1/E1 SPA" and Chapter 19, "Configuring the 2-Port and 4-Port Channelized T3 SPAs" were modified to clarify support of MLPPP and MLFR for both E1 and T1 links. • Added cRTP to the supported features list for the serial SPAs in Chapter 16, "Overview of the Serial SPAs." • Document was modified with the following updates in Chapter 4, "Configuring the SIPs and SSC": – Removed references to support of software-based MLFR. – In the "Assigning an Interface to an MLPPP Bundle," moved step order of the **ppp multilink** command and qualified it as optional. – Under "MLPPP Configuration Guidelines," added guidelines for distributed links on the Cisco 7600 SIP-200 and restrictions. – Under "MLPPP Configuration Tasks" and "MLFR Configuration Tasks," added task to emphasize that distributed CEF is required for these features; however, dCEF is automatically enabled on the Cisco 7600 series router.
12.2(18)SXE2	OL-5070-02	July 25, 2005	Second release. The Cisco 7600 SSC-400 and IPSec VPN SPA were introduced.
12.2(18)SXE	OL-5070-01	March 28, 2005	First release.

Organization

This document contains the following chapters:

Chapter	Title	Description
Chapter 1	Using Cisco IOS Software	Provides an introduction to accessing the command-line interface (CLI) and using the Cisco IOS software and related tools.
Chapter 2	SIP, SSC, and SPA Product Overview	Provides a brief introduction to the SIP and SPA products on the Cisco 7600 series router, and information about SIP, SSC, SPA, and optics compatibility.
Chapter 3	Overview of the SIPs and SSC	Describes release history, and feature and Management Information Base (MIB) support for the SIPs and SSCs on the Cisco 7600 series router.
Chapter 4	Configuring the SIPs and SSC	Describes related configuration and verification information for the SIPs and SSCs on the Cisco 7600 series router.
Chapter 5	Troubleshooting the SIPs and SSC	Describes techniques that you can use to troubleshoot the operation of the SIPs and SSCs on the Cisco 7600 series router.
Chapter 6	Overview of the ATM SPAs	Describes release history, feature and Management Information Base (MIB) support, and an introduction to the ATM SPA architecture on the Cisco 7600 series router.
Chapter 7	Configuring the ATM SPAs	Describes the related configuration and verification information for the ATM SPAs on the Cisco 7600 series router.
Chapter 8	Troubleshooting the ATM SPAs	Describes techniques that you can use to troubleshoot the operation of the ATM SPAs on the Cisco 7600 series router.
Chapter 9	Overview of the CEoP and Channelized ATM SPAs	Describes release history, feature and Management Information Base (MIB) support, and an introduction to the CEoP SPA architecture on the Cisco 7600 series router.
Chapter 10	Configuring the CEoP and Channelized ATM SPAs	Describes the related configuration and verification information for the CEoP and Channelized SPAs on the Cisco 7600 series router.
Chapter 11	Overview of the Ethernet SPAs	Describes release history, feature and Management Information Base (MIB) support, and an introduction to the Gigabit Ethernet SPA architecture on the Cisco 7600 series router.
Chapter 12	Configuring the Fast Ethernet and Gigabit Ethernet SPAs	Describes the related configuration and verification information for the Gigabit Ethernet SPAs on the Cisco 7600 series router.

Chapter	Title	Description
Chapter 13	Troubleshooting the Fast Ethernet and Gigabit Ethernet SPAs	Describes techniques that you can use to troubleshoot the operation of the Gigabit Ethernet SPAs on the Cisco 7600 series router.
Chapter 14	Overview of the POS SPAs	Describes release history, feature and Management Information Base (MIB) support, and an introduction to the POS SPA architecture on the Cisco 7600 series router.
Chapter 15	Configuring the POS SPAs	Describes the related configuration and verification information for the POS SPAs on the Cisco 7600 series router.
Chapter 16	Overview of the Serial SPAs	Describes release history, feature and Management Information Base (MIB) support, and an introduction to the serial SPA architecture on the Cisco 7600 series router.
Chapter 17	Configuring the 8-Port Channelized T1/E1 SPA	Describes the related configuration and verification information for the 8-Port Channelized T1/E1 SPAs on the Cisco 7600 series router.
Chapter 18	Configuring the 2-Port and 4-Port Clear Channel T3/E3 SPAs	Describes the related configuration and verification information for the 2-Port and 4-Port Clear Channel T3/E3 SPAs on the Cisco 7600 series router.
Chapter 19	Configuring the 2-Port and 4-Port Channelized T3 SPAs	Describes the related configuration and verification information for the 2-Port and 4-Port Channelized T3 SPAs on the Cisco 7600 series router.
Chapter 20	Configuring the 1-Port Channelized OC-3/STM-1 SPA	Describes the related configuration and verification information for the 1-Port Channelized OC-3/STM-1 SPA on the Cisco 7600 series router.
Chapter 21	Troubleshooting the Serial SPAs	Describes techniques that you can use to troubleshoot the operation of the serial SPAs on the Cisco 7600 series router.
Chapter 22	Overview of the IPSec VPN SPA	Describes release history, feature and Management Information Base (MIB) support, and an introduction to the IPSec VPN SPA architecture on the Cisco 7600 series router.
Chapter 23	Configuring VPNs on the IPSec VPN SPA	Describes the related configuration and verification information for IPSec VPNs using the IPSec VPN SPA on the Cisco 7600 series router.
Chapter 24	Configuring IKE Features Using the IPSec VPN SPA	Describes the related configuration and verification information for Internet Key Exchange (IKE) features using the IPSec VPN SPA on the Cisco 7600 series router.
Chapter 25	Configuring Enhanced IPSec Features Using the IPSec VPN SPA	Describes the related configuration and verification information for enhanced IPSec features using the IPSec VPN SPA on the Cisco 7600 series router.

Chapter	Title	Description
Chapter 26	Configuring PKI Using the IPSec VPN SPA	Describes the related configuration and verification information for Public Key Infrastructure (PKI) features using the IPSec VPN SPA on the Cisco 7600 series router.
Chapter 27	Configuring Advanced VPNs Using the IPSec VPN SPA	Describes the related configuration and verification information for advanced IPSec VPNs using the IPSec VPN SPA on the Cisco 7600 series router.
Chapter 28	Configuring Duplicate Hardware Configurations and IPSec Failover Using the IPSec VPN SPA	Describes the related configuration and verification information for duplicate hardware configurations and IPSec failover using the IPSec VPN SPA on the Cisco 7600 series router.
Chapter 29	Configuring Monitoring and Accounting for the IPSec VPN SPA	Describes the related configuration and verification information for monitoring and accounting using the IPSec VPN SPA on the Cisco 7600 series router.
Chapter 30	Troubleshooting the IPSec VPN SPA	Describes techniques that you can use to troubleshoot the operation of the IPSec VPN SPA on the Cisco 7600 series router.
Chapter 31	Upgrading Field-Programmable Devices	Provides information about upgrading the field-programmable devices on the Cisco 7600 series router.

Related Documentation

This section refers you to other documentation that also might be useful as you configure your Cisco 7600 series router. The documentation listed below is available online.

Cisco 7600 Series Router Documentation

As you configure SIPs and SPAs on your Cisco 7600 series router, you should also refer to the following companion publication for important hardware installation information:

- *Cisco 7600 Series Router SIP, SSC, and SPA Hardware Installation Guide*

An overview of the Cisco 7600 series router features, benefits, and applications can be found in the *Cisco 7600 Series Internet Router Essentials* document located at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/products_quick_start09186a0080092248.html

Some of the following other Cisco 7600 series router publications might be useful to you as you configure your Cisco 7600 series router.

- *Cisco 7600 Series Cisco IOS Software Configuration Guide*

  http://www.cisco.com/en/US/products/hw/routers/ps368/products_installation_and_configuration_guides_list.html

- *Cisco 7600 Series Cisco IOS Command Reference*

  http://www.cisco.com/en/US/products/hw/routers/ps368/prod_command_reference_list.html

- *Cisco 7600 Series Cisco IOS System Message Guide*

  http://www.cisco.com/en/US/products/hw/routers/ps368/products_system_message_guides_list.html
- *Cisco 7600 Series Internet Router MIB Specifications Guide*

  http://www.cisco.com/en/US/products/hw/routers/ps368/prod_technical_reference_list.html

Several other publications are also related to the Cisco 7600 series router. For a complete reference of related documentation, refer to the *Cisco 7600 Series Routers Documentation Roadmap* located at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/products_documentation_roadmaps_list.html

Other Cisco IOS Software Publications

Your router and the Cisco IOS software running on it contain extensive features. You can find documentation for Cisco IOS software features at the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/tsd_products_support_category_home.html

Cisco IOS Release 12.2SX Software Publications

Documentation for Cisco IOS Release 12.2SX, including command reference and system error messages, can be found at the following URL:

http://www.cisco.com/en/US/products/ps6017/tsd_products_support_series_home.html

Cisco IOS Release 12.2SR Software Publications

Documentation for Cisco IOS Release 12.2SR, including command reference and system error messages, can be found at the following URL:

http://www.cisco.com/en/US/products/ps6922/tsd_products_support_series_home.html

Document Conventions

Within the SIP and SPA software configuration guides, the term *router* is generally used to refer to a variety of Cisco products (for example, routers, access servers, and switches). Routers, access servers, and other networking devices that support Cisco IOS software are shown interchangeably within examples. These products are used only for illustrative purposes; that is, an example that shows one product does not necessarily indicate that other products are not supported.

This documentation uses the following conventions:

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

Command syntax descriptions use the following conventions:

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

Nested sets of square brackets or braces indicate optional or required choices within optional or required elements. For example:

<table>
<tr><th>Convention</th><th>Description</th></tr>
<tr><td>[x {y | z}]</td><td>Braces and a vertical line within square brackets indicate a required choice within an optional element.</td></tr>
</table>

Examples use the following conventions:

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

The following conventions are used to attract the attention of the reader:

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

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

Tip Means *the following information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.

Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, 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

Part 1

Introduction

CHAPTER 1

Using Cisco IOS Software

This chapter provides information to prepare you to configure a SPA interface processor (SIP) or shared port adapter (SPA) using the Cisco IOS software. It includes the following sections:

Accessing the CLI Using a Router Console

The following sections describe how to access the command-line interface (CLI) using a directly-connected console or by using Telnet or a modem to obtain a remote console:

For more detailed information about configuring and accessing a router through various services, refer to the *Cisco IOS Terminal Services Configuration Guide* and *Cisco IOS Terminal Services Command Reference* publications.

For more information about making the console cable connections, refer to the *Cisco 7600 Series Router Module Installation Guide*.

Accessing the CLI Using a Directly-Connected Console

This section describes how to connect to the console port on the router and use the console interface to access the CLI.

The console port on a Cisco 7600 series router is an EIA/TIA-232 asynchronous, serial connection with hardware flow control and an RJ-45 connector. The console port is located on the front panel of the supervisor engine, as shown in Figure 1-1 and Figure 1-2.

Figure 1-1 Supervisor Engine 720 Console Port Connector

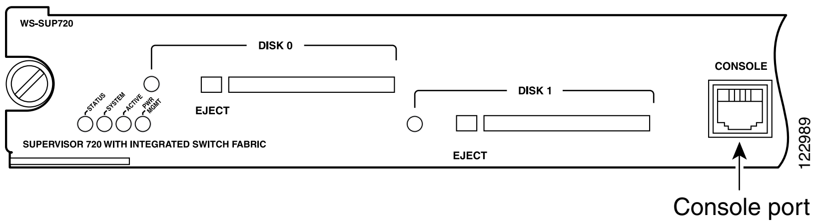
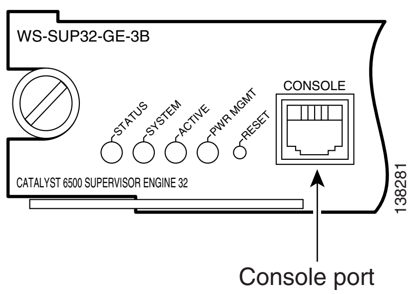


Figure 1-2 Supervisor Engine 32 Console Port Connector



Connecting to the Console Port

Before you can use the console interface on the router using a terminal or PC, you must perform the following steps:

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

- 9600 bits per second (bps)
- 8 data bits
- No parity
- 2 stop bits

Note These are the default serial communication parameters on the router. For information about how to change the default settings to meet the requirements of your terminal or host, refer to the *Cisco IOS Terminal Services Configuration Guide*.

Step 2 Connect a terminal or PC to the console port using *one* of the following methods:

a. To connect to the console port using the cable and adapters provided in the accessory kit that shipped with your Cisco 7600 series router:

– Place the console port mode switch in the in position (factory default).

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

b. To connect to the console port using a Catalyst 5000 family Supervisor Engine III console cable:

- Place the console port mode switch in the out position.
- Connect to the port using the Supervisor Engine III cable and the appropriate adapter for the terminal connection.

Using the Console Interface

To access the CLI using the console interface, complete the following steps:

Step 1 After you attach the terminal hardware to the console port on the router and you configure your terminal emulation software with the proper settings, the following prompt appears:

```
Press Return for Console prompt
```

Step 2 Press **Return** to enter user EXEC configuration mode. The following prompt appears:

```
Router>
```

Step 3 From user EXEC configuration mode, enter the **enable** command as shown in the following example:

```
Router> enable
```

Step 4 At the password prompt, enter your system's password. (The following example shows entry of the password called "enablepass"):

```
Password: enablepass
```

Step 5 When your enable password is accepted, the privileged EXEC configuration mode prompt appears:

```
Router#
```

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

Step 7 To exit the console session, enter the **quit** command as shown in the following example:

```
Router# quit
```

Accessing the CLI from a Remote Console Using Telnet

This section describes how to connect to the console interface on a router using Telnet to access the CLI.

Preparing to Connect to the Router Console Using Telnet

Before you can access the router remotely using Telnet from a TCP/IP network, you need to configure the router to support virtual terminal lines (vtys) using the **line vty** global configuration command. You also should configure the vtys to require login and specify a password.

Note To prevent disabling login on the line, be careful that you specify a password with the **password** command when you configure the **login** line configuration command. If you are using authentication, authorization, and accounting (AAA), you should configure the **login authentication** line configuration command. To prevent disabling login on the line for AAA authentication when you configure a list with the **login authentication** command, you must also configure that list using the **aaa authentication login** global configuration command. For more information about AAA services, refer to the *Cisco IOS Security Configuration Guide* and *Cisco IOS Security Command Reference* publications.

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

Using Telnet to Access a Console Interface

To access a console interface using Telnet, complete the following steps:

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

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

In this syntax, *host* is the router host name or an IP address, *port* is a decimal port number (23 is the default), and *keyword* is a supported keyword. For more information, refer to the *Cisco IOS Terminal Services Command Reference*.

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

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

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

Step 2 At the password prompt, enter your login password. The following example shows entry of the password called “mypass”:

```
User Access Verification

Password: mypass
```

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

Step 3 From user EXEC configuration mode, enter the **enable** command as shown in the following example:

```
Router> enable
```

Step 4 At the password prompt, enter your system's password. (The following example shows entry of the password called "enablepass"):

```
Password: enablepass
```

Step 5 When the enable password is accepted, the privileged EXEC configuration mode prompt appears:

```
Router#
```

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

Step 7 To exit the Telnet session, use the **exit** or **logout** command as shown in the following example:

```
Router# logout
```

Accessing the CLI from a Remote Console Using a Modem

To access the router remotely using a modem through an asynchronous connection, connect the modem to the console port.

The console port on a Cisco 7600 series router is an EIA/TIA-232 asynchronous, serial connection with hardware flow control and an RJ-45 connector. The console port is located on the front panel of the supervisor engine, as shown in Figure 1-3 and Figure 1-4.

Figure 1-3 Supervisor Engine 720 Console Port Connector

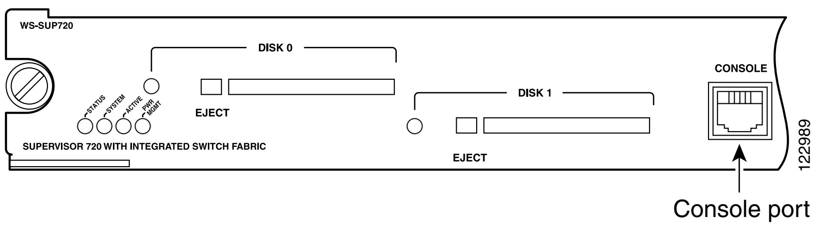
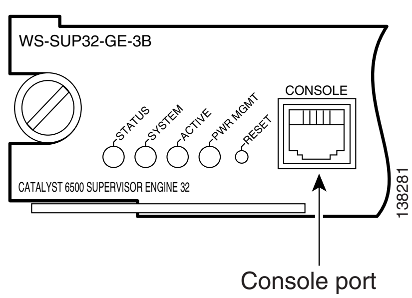


Figure 1-4 Supervisor Engine 32 Console Port Connector



To connect a modem to the console port, place the console port mode switch in the in position. Connect to the port using the RJ-45-to-RJ-45 cable and the RJ-45-to-DB-25 DCE adapter (labeled "Modem").

Using Keyboard Shortcuts

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

Table 1-1 lists the keyboard shortcuts for entering and editing commands.

Table 1-1 Keyboard Shortcuts

Keystrokes	Purpose
Ctrl-B or the **Left Arrow** key[1]	Move the cursor back one character
Ctrl-F or the **Right Arrow** key1	Move the cursor forward one character
Ctrl-A	Move the cursor to the beginning of the command line
Ctrl-E	Move the cursor to the end of the command line
Esc B	Move the cursor back one word
Esc F	Move the cursor forward one word

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

Using the History Buffer to Recall Commands

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

Table 1-2 lists the history substitution commands.

Table 1-2 History Substitution Commands

Command	Purpose
Ctrl-P or the **Up Arrow** key[1]	Recall 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** key1	Return to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the **Up Arrow** key.
`Router# show history`	While in EXEC mode, list the last several commands you have just entered.

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

Understanding Command Modes

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

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

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

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

Table 1-3 describes how to access and exit various common command modes of the Cisco IOS software. It also shows examples of the prompts displayed for each mode.

Table 1-3 Accessing and Exiting Command Modes

Command Mode	Access Method	Prompt	Exit Method
User EXEC	Log in.	`Router>`	Use the **logout** command.
Privileged EXEC	From user EXEC mode, use the **enable** EXEC command.	`Router#`	To return to user EXEC mode, use the **disable** command.
Global configuration	From privileged EXEC mode, use the **configure terminal** privileged EXEC command.	`Router(config)#`	To return to privileged EXEC mode from global configuration mode, use the **exit** or **end** command.
Interface configuration	From global configuration mode, specify an interface using an **interface** command.	`Router(config-if)#`	To return to global configuration mode, use the **exit** command. To return to privileged EXEC mode, use the **end** command.
ROM monitor	From privileged EXEC mode, use the **reload** EXEC command. Press the **Break** key during the first 60 seconds while the system is booting.	`>`	To exit ROM monitor mode, use the **continue** command.

For more information on command modes, refer to the “Using the Command-Line Interface” chapter in the *Cisco IOS Configuration Fundamentals and Network Management Configuration Guide*.

Getting Help

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

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

Table 1-4 Help Commands and Purpose

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

Finding Command Options Example

This section provides an example of how to display syntax for a command. The syntax can consist of optional or required keywords and arguments. To display keywords and arguments for a command, enter a question mark (**?**) at the configuration prompt or after entering part of a command followed by a space. The Cisco IOS software displays a list and brief description of available keywords and arguments. For example, if you were in global configuration mode and wanted to see all the keywords or arguments for the **arap** command, you would type **arap ?**.

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

Table 1-5 shows examples of how you can use the question mark (**?**) to assist you in entering commands.

Table 1-5 Finding Command Options

Command	Comment
`Router>` **`enable`** `Password:` *`<password>`* `Router#`	Enter the **enable** command and password to access privileged EXEC commands. You are in privileged EXEC mode when the prompt changes to a "`#`" from the "`>`"; for example, `Router>` to `Router#`.

`Router#` **`configure terminal`**
`Enter configuration commands, one per line. End with CNTL/Z.`
`Router(config)#`
	Enter the **configure terminal** privileged EXEC command to enter global configuration mode. You are in global configuration mode when the prompt changes to `Router(config)#`.

`Router(config)#` **`interface serial ?`**
`  <0-6>     Serial interface number`
`Router(config)#` **`interface serial 4 ?`**
`  /`
`Router(config)#` **`interface serial 4/ ?`**
`  <0-3>     Serial interface number`
`Router(config)#` **`interface serial 4/0 ?`**
`<cr>`
`Router(config)#` **`interface serial 4/0`**
`Router(config-if)#`
	Enter interface configuration mode by specifying the serial interface that you want to configure using the **interface serial** global configuration command.
	
	Enter **?** to display what you must enter next on the command line. In this example, you must enter the serial interface slot number and port number, separated by a forward slash.
	
	When the <cr> symbol is displayed, you can press **Enter** to complete the command.
	
	You are in interface configuration mode when the prompt changes to `Router(config-if)#`.

Table 1-5 Finding Command Options (continued)

<table>
<tr><th>Command</th><th>Comment</th></tr>
<tr><td><pre>
Router(config-if)# ?
Interface configuration commands:
  .
  .
  .
  ip   Interface Internet Protocol config commands
  keepalive   Enable keepalive
  lan-name   LAN Name command
  llc2   LLC2 Interface Subcommands
  load-interval   Specify interval for load calculation for an
                        interface
  locaddr-priority   Assign a priority group
  logging   Configure logging for interface
  loopback   Configure internal loopback on an interface
  mac-address   Manually set interface MAC address
  mls   mls router sub/interface commands
  mpoa   MPOA interface configuration commands
  mtu   Set the interface Maximum Transmission Unit (MTU)
  netbios   Use a defined NETBIOS access list or enable
                        name-caching
  no   Negate a command or set its defaults
  nrzi-encoding   Enable use of NRZI encoding
  ntp   Configure NTP
  .
  .
  .
Router(config-if)#
</pre></td><td>Enter ? to display a list of all the interface configuration commands available for the serial interface. This example shows only some of the available interface configuration commands.</td></tr>
<tr><td><pre>
Router(config-if)# ip ?
Interface IP configuration subcommands:
  access-group   Specify access control for packets
  accounting   Enable IP accounting on this interface
  address   Set the IP address of an interface
  authentication   authentication subcommands
  bandwidth-percent   Set EIGRP bandwidth limit
  broadcast-address   Set the broadcast address of an interface
  cgmp   Enable/disable CGMP
  directed-broadcast   Enable forwarding of directed broadcasts
  dvmrp   DVMRP interface commands
  hello-interval   Configures IP-EIGRP hello interval
  helper-address   Specify a destination address for UDP broadcasts
  hold-time   Configures IP-EIGRP hold time
  .
  .
  .
Router(config-if)# ip
</pre></td><td>Enter the command that you want to configure for the interface. This example uses the ip command.

Enter ? to display what you must enter next on the command line. This example shows only some of the available interface IP configuration commands.</td></tr>
</table>

Table 1-5 Finding Command Options (continued)

Command	Comment

`Router(config-if)# ip address ?`
`  A.B.C.D             IP address`
`  negotiated          IP Address negotiated over PPP`
`Router(config-if)# ip address`
	Enter the command that you want to configure for the interface. This example uses the **ip address** command.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter an IP address or the **negotiated** keyword.
	A carriage return (<cr>) is not displayed; therefore, you must enter additional keywords or arguments to complete the command.

`Router(config-if)# ip address 172.16.0.1 ?`
`  A.B.C.D             IP subnet mask`
`Router(config-if)# ip address 172.16.0.1`
	Enter the keyword or argument that you want to use. This example uses the 172.16.0.1 IP address.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter an IP subnet mask.
	A <cr> is not displayed; therefore, you must enter additional keywords or arguments to complete the command.

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

`Router(config-if)# ip address 172.16.0.1 255.255.255.0` `Router(config-if)#`	In this example, **Enter** is pressed to complete the command.

Using the no and default Forms of Commands

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

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

Saving Configuration Changes

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

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

It might take a minute or two to save the configuration. After the configuration has been saved, the following output appears:

```
[OK]
Router#
```

On most platforms, this task saves the configuration to NVRAM. On the Class A Flash file system platforms, this task saves the configuration to the location specified by the CONFIG_FILE environment variable. The CONFIG_FILE variable defaults to NVRAM.

Filtering Output from the show and more Commands

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

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

show *command* | {**begin** | **include** | **exclude**} *regular-expression*

The output matches certain lines of information in the configuration file. The following example illustrates how to use output modifiers with the **show interface** command when you want the output to include only lines in which the expression "protocol" appears:

```
Router# show interface | include protocol

FastEthernet0/0 is up, line protocol is up
Serial4/0 is up, line protocol is up
Serial4/1 is up, line protocol is up
Serial4/2 is administratively down, line protocol is down
Serial4/3 is administratively down, line protocol is down
```

For more information on the search and filter functionality, refer to the "Using the Command-Line Interface" chapter in the *Cisco IOS Configuration Fundamentals and Network Management Configuration Guide*.

Finding Support Information for Platforms and Cisco Software Images

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

Using Cisco Feature Navigator

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://tools.cisco.com/ITDIT/CFN/jsp/index.jsp. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Using Software Advisor

To see if a feature is supported by a Cisco IOS release, to locate the software document for that feature, or to check the minimum software requirements of Cisco IOS software with the hardware installed on your router, Cisco maintains the Software Advisor tool on Cisco.com at http://www.cisco.com/cgi-bin/Support/CompNav/Index.pl.

You must be a registered user on Cisco.com to access this tool.

Using Software Release Notes

Cisco IOS software releases include release notes that provide the following information:

- Platform support information
- Memory recommendations
- New feature information
- Open and resolved severity 1 and 2 caveats for all platforms

Release notes are intended to be release-specific for the most current release, and the information provided in these documents may not be cumulative in providing information about features that first appeared in previous releases. Refer to Cisco Feature Navigator for cumulative feature information.

CHAPTER 2

SIP, SSC, and SPA Product Overview

This chapter provides an introduction to SPA interface processors (SIPs), SPA services cards (SSCs), and shared port adapters (SPAs). It includes the following sections:

For more hardware details for the specific SIPs, SSCs, and SPAs that are supported on the Cisco 7600 series router, refer to the companion publication, Cisco 7600 Series Router SIP, SSC, and SPA Hardware Installation Guide.

Introduction to SIPs, SSCs, and SPAs

SIPs, SSCs, and SPAs are a new carrier card and port adapter architecture to increase modularity, flexibility, and density across Cisco Systems routers for network connectivity. This section describes the SIPs, SSCs, and SPAs and provides some guidelines for their use.

SPA Interface Processors

The following list describes some of the general characteristics of a SIP:

- A SIP is a carrier card that inserts into a router slot like a line card. It provides no network connectivity on its own.
- A SIP contains one or more subslots, which are used to house one or more SPAs. The SPA provides interface ports for network connectivity.
- During normal operation the SIP should reside in the router fully populated either with functional SPAs in all subslots, or with a blank filler plate (SPA-BLANK=) inserted in all empty subslots.
- SIPs support online insertion and removal (OIR) with SPAs inserted in their subslots. SPAs also support OIR and can be inserted or removed independently from the SIP.

SPA Services Cards

The following list describes some of the general charateristics of an SSC:

- An SSC is a carrier card that inserts into a router slot like a line card. It provides no network connectivity.
- An SSC provides one or more subslots, which are used to house one or more SPAs. The supported SPAs do not provide interface ports for network connectivity, but provide certain services.
- During normal operation the SSC should reside in the router fully populated either with functional SPAs in all subslots, or with a blank filler plate (SPA-BLANK=) inserted in all empty subslots.
- SSCs support online insertion and removal (OIR) with SPAs inserted in their subslots. SPAs also support OIR and can be inserted or removed independently from the SSC.

Shared Port Adapters

The following list describes some of the general characteristics of a SPA:

- A SPA is a modular type of port adapter that inserts into a subslot of a compatible SIP carrier card to provide network connectivity and increased interface port density. A SIP can hold one or more SPAs, depending on the SIP type.
- Some SPAs provide services rather than network connectivity, and insert into subslots of compatible SSCs. For example, the IPSec VPN SPA provides services such as IP Security (IPSec) encryption/decryption, generic routing encapsulation (GRE), and Internet Key Exchange (IKE) key generation.
- SPAs are available in the following sizes, as shown in Figure 2-1 and Figure 2-2:
  - Single-height SPA—Inserts into one SIP subslot.
  - Double-height SPA—Inserts into two single, vertically aligned SIP subslots.

Figure 2-1 Single-Height and Double-Height SPA Sizes

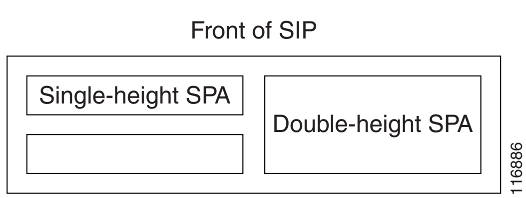
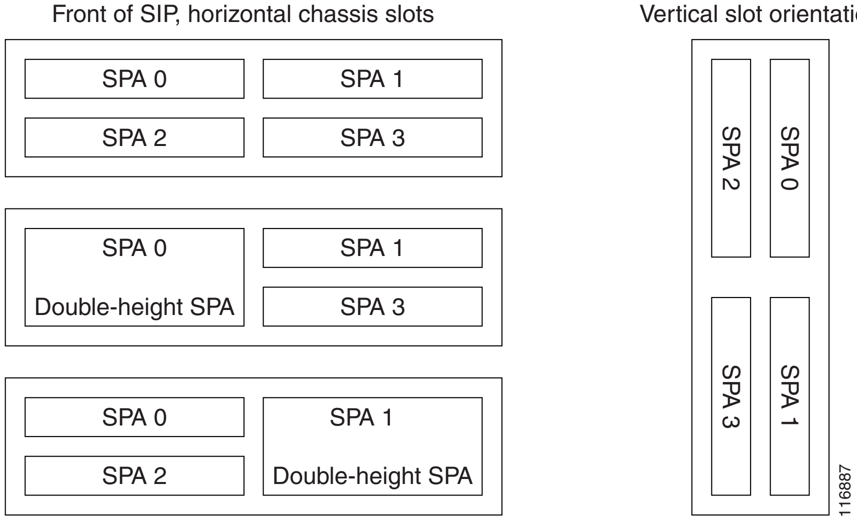


Figure 2-2 Horizontal and Vertical Chassis Slot Orientation for SPAs



- Each SPA provides a certain number of connectors, or ports, that are the interfaces to one or more networks. These interfaces can be individually configured using the Cisco IOS command-line interface (CLI).
- Either a blank filler plate or a functional SPA should reside in every subslot of an SIP during normal operation to maintain cooling integrity. Blank filler plates are available in single-height form only.
- SPAs support online insertion and removal (OIR). They can be inserted or removed independently from the SIP. SIPs also support online insertion and removal (OIR) with SPAs inserted in their subslots.

SIP, SSC, and SPA Compatibility

The following tables show SIP and SPA compatibility by SPA technology area on the Cisco 7600 series router.

Note For more information about the introduction of support for different SIPs and SPAs, refer to the "Release History" sections in the overview chapters of the Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide.

Table 2-1 SIP and SPA Compatibility Table for ATM SPAs

SPA	Product ID	SIP Type			
		Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600	Cisco 7600 SSC-400
2-Port and 4-Port OC-3c/STM-1 ATM SPA	SPA-2XOC3-ATM, SPA-4XOC3-ATM	Yes	Yes	No	No
1-Port OC-12c/STM-4 ATM SPA	SPA-1XOC12-ATM	No	Yes	No	No
1-Port OC-48c/STM-16 ATM SPA	SPA-1XOC48-ATM	No	Yes	No	No

Table 2-2 SIP and SPA Compatibility Table for Ethernet SPAs

SPA	Product ID	SIP Type			
		Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600	Cisco 7600 SSC-400
1-Port 10-Gigabit Ethernet SPA[1]	SPA-1XTENGE-XENP, SPA-1XTENGE-XFP,	No	No	Yes	No
	SPA-1X10GE-L-V2	No	Yes	Yes	No
2-Port Gigabit Ethernet SPA	SPA-2X1GE, SPA-2X1GE-V2	No	Yes	No	No
5-Port Gigabit Ethernet SPA	SPA-5X1GE	No	No	Yes	No
	SPA-5X1GE-V2	No	Yes	Yes	No
10-Port Gigabit Ethernet SPA	SPA-10X1GE, SPA-10X1GE-V2	No	No	Yes	No
4-Port and 8-Port Fast Ethernet SPA	SPA-4X1FE-TX-V2, SPA-8X1FE-TX-V2	Yes	No	No	No

1. Only one 1-Port 10-Gigabit Ethernet SPAcan be installed in a SIP-400 at a time; no other SPAs can be installed in the same SIP-400.

Table 2-3 SIP and SPA Compatibility Table for the IPSec VPN SPA

SPA	Product ID	SIP Type			
		Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600	Cisco 7600 SSC-400
IPSec VPN SPA	SPA-IPSEC-2G	No	No	No	Yes

Table 2-4 SIP and SPA Compatibility Table for POS SPAs

SPA	Product ID	SIP Type			
		Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600	Cisco 7600 SSC-400
2-Port and 4-Port OC-3c/STM-1 POS SPA	SPA-2XOC3-POS, SPA-4XOC3-POS	Yes	Yes	No	No
1-Port OC-12c/STM-4 POS SPA	SPA-1XOC12-POS	No	Yes	No	No
1-Port OC-48c/STM-16 POS SPA	SPA-1XOC48-POS/RPR	No	Yes	No	No
2-Port and 4-Port OC-48c/STM-16 POS SPA	SPA-2XOC48-POS/RPR, SPA-4XOC48-POS/RPR	No	No	Yes	No
1-Port OC-192c/STM-64 POS/RPR SPA	SPA-OC192POS-LR, SPA-OC192POS-VSR, SPA-OC192POS-XFP	No	No	Yes	No

Table 2-5 SIP and SPA Compatibility Table for Serial SPAs

SPA	Product ID	SIP Type			
		Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600	Cisco 7600 SSC-400
1-Port Channelized OC-3/STM-1 SPA	SPA-1XCHSTM1/OC3	Yes	Yes	No	No
2-Port and 4-Port Channelized T3 SPA	SPA-2XCT3/DS0, SPA-4XCT3/DS0	Yes	Yes	No	No
2-Port and 4-Port Clear Channel T3/E3 SPA	SPA-2XT3/E3, SPA-4XT3/E3	Yes	Yes	No	No
8-Port Channelized T1/E1 SPA	SPA-8XCHT1/E1	Yes	Yes	No	No

Table 2-6 SIP and SPA Compatibility Table for CEoP SPAs

SPA	Product ID	SIP Type			
		Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600	Cisco 7600 SSC-400
1-Port Channelized OC-3/STM-1 ATM CEoP SPA	SPA-1CHOC3-CE-ATM	No	Yes	No	No
24-Port Channelized T1/E1 ATM CEoP SPA	SPA-24CHT1-CE-ATM	No	Yes	No	No
2-Port Channelized T3/E3 ATM CEoP SPA	SPA-2CHT3-CE-ATM	No	Yes	No	No

Modular Optics Compatibility

Some SPAs implement small form-factor pluggable (SFP) optical transceivers to provide network connectivity. An SFP module is a transceiver device that mounts into the front panel to provide network connectivity.

Cisco Systems qualifies the SFP modules that can be used with SPAs.

Note The SPAs will only accept the SFP modules listed as supported in this document. An SFP check is run every time an SFP module is inserted into a SPA and only SFP modules that pass this check will be usable.

Table 2-7 shows the types of optics modules that have been qualified for use with a SPA:

Table 2-7 SPA Optics Compatibility

SPA	Qualified Optics Modules (Cisco Part Numbers)
2-Port and 4-Port OC-3c/STM-1 ATM SPA	• SFP-OC3-MM • SFP-OC3-SR • SFP-OC3-IR1 • SFP-OC3-LR1 • SFP-OC3-LR2
1-Port OC-12c/STM-4 ATM SPA	• SFP-OC12-MM • SFP-OC12-SR • SFP-OC12-IR1 • SFP-OC12-LR1 • SFP-OC12-LR2
1-Port OC-48c/STM-16 ATM SPA	• SFP-OC48-IR1 • SFP-OC48-SR
1-Port 10-Gigabit Ethernet SPA	• XFP-10GLR-OC192SR • XFP-10GER-OC192IR • XFP-10GZR-OC192LR
2-Port Gigabit Ethernet SPA	• SFP-GE-S • SFP-GE-L • SFP-GE-Z • SFP-GE-T
5-Port Gigabit Ethernet SPA	• SFP-GE-S • SFP-GE-L • SFP-GE-Z • SFP-GE-T

Table 2-7 SPA Optics Compatibility (continued)

SPA	Qualified Optics Modules (Cisco Part Numbers)
10-Port Gigabit Ethernet SPA	• SFP-GE-S • SFP-GE-L • SFP-GE-Z • SFP-GE-T
2-Port and 4-Port OC-3c/STM-1 POS SPA	• SFP-OC3-MM • SFP-OC3-SR • SFP-OC3-IR1 • SFP-OC3-LR1 • SFP-OC3-LR2
1-Port OC-12c/STM-4 POS SPA	• SFP-OC12-MM • SFP-OC12-SR • SFP-OC12-IR1 • SFP-OC12-LR1 • SFP-OC12-LR2
1-Port OC-48c/STM-16 POS SPA	• SFP-OC48-SR • SFP-OC48-IR1 • SFP-OC48-LR2
2-Port and 4-Port OC-48c/STM-16 POS SPA	• SFP-OC48-SR • SFP-OC48-IR1 • SFP-OC48-LR2
1-Port OC-192c/STM-64 POS/RPR XFP SPA	• XFP-10GLR-OC192SR • XFP-10GER-OC192IR • XFP-10GZR-OC192LR
1-Port Channelized OC-3/STM-1 SPA	• SFP-OC3-SR • SFP-OC3-IR1 • SFP-OC3-LR1 • SFP-OC3-LR2
1-Port Channelized OC-3/STM-1 ATM CEoP SPA	• SFP-OC3-MM • SFP-OC3-SR • SFP-OC3-IR1 • SFP-OC3-LR1 • SFP-OC3-LR2

PART 2

SPA Interface Processors and SPA Services Cards

CHAPTER 3

Overview of the SIPs and SSC

This chapter provides an overview of the release history, and feature and Management Information Base (MIB) support for the Cisco 7600 SIP-200, Cisco 7600 SIP-400, Cisco 7600 SIP-600, and Cisco 7600 SSC-400.

This chapter includes the following sections:

Release History

Note For release history information about the introduction of SPA support on the SIPs, refer to the corresponding "Overview" chapters in the SPA technology sections of this document. In addition, features specific to certain SPA technologies are documented in the corresponding SPA sections of this document.

Release	Modification
Cisco IOS Release 12.2(33)SRC	Support for the following features was introduced: • CT3 CEoP on c7600-SIP-400 • Accelerated Lawful Intercept on Cisco 7600 SIP-400 • CoPP Enhancements of Cisco 7600 SIP-400 • PPPoEoE on Cisco 7600 SIP-400 • Source IPv4 and Source MAC Address Binding on Cisco 7600 SIP-400 • 12in1 Serial SPA support on 7600/SIP200 • IMA on SIP-400 for 24xT1/E1 CEOP and 1xOC3 CEOP SPAs • IGMP Snooping support on SIP-200 • AFC and PFC support on Multilink Interface on SIP-200 for 2- and 4-port CT3, 8-port channelized T1/E1 channelized, 1-port channelized OC3/STM-1 SPAs • Programmable BERT patterns enhancement on SIP-200 for 2- and 4-port channelized T3 and 1-port channelized OC3/STM-1 SPAs • TDM Local switching • Phase 2 Access Circuit Redundancy with Local Switching • SPA-1xCHSTM1/OC3 • Cisco Channelized T3 to DS0 Shared Port Adapter (SPA-2XCT3/DS0, SPA-4XCT3/DS0) • Cisco 8-Port Channelized T1/E1 Shared Port Adapter (SPA-8XCHT1/E1) • Cisco Clear Channel T3/E3 Shared Port Adapter (SPA-2XT3/E3, SPA-4XT3/E3)
Cisco IOS Release 12.2(33)SRB1	Support for the following feature was introduced: • MTU support on MLPPP interfaces on a Cisco 7600 SIP-200 • Any Transport over MPLS over GRE (AToM over GRE) on a Cisco 7600 SIP-400
Cisco IOS Release 12.2(33)SRB	Support for the following features was introduced: • Software-based MLP bundles from 256 to 1024 on a Cisco 7600 SIP-200 • Lawful Intercept on a Cisco 7600 SIP-400 • Per-subscriber/per-protocol CoPP support on a Cisco 7600 SIP-400 • Security ACLs on a Cisco 7600 SIP-400 • Percent priority/percent bandwidth support on a Cisco 7600 SIP-400 • Network Clock Support on a Cisco 7600 SIP-200 • IGMP/PIM snooping for VPLS pseudowire on a Cisco 7600 SIP-400 • Dual-priority queue support on a Cisco 7600 SIP-400

Cisco IOS Release 12.2(33)SRA	Support for the following features was introduced on the Cisco 7600 SIP-200: • Bridge Control Protocol (BCP) over dMLPPP • MPLS over RBE • Multi-VC to VLAN Scalability • QoS support on bridging features • Software-based dMLPPP • Software-based dMLFR • Tag-Native Mode for Trunk BCP Support for the following features was introduced on the Cisco 7600 SIP-400: • Ethernet over MPLS (EoMPLS) VC Scaling • Ingress/Egress CoS classification with ingress policing per VLAN or EoMPLS VC • Hierarchical VPLS (H-VPLS) with MPLS Edge • Hierarchical QoS support for Ethernet over MPLS (EoMPLS) VCs • Multipoint Bridging (MPB) • Multi-VC to VLAN scalability • Multi-VLAN to VC support • QoS support on bridging features • Tag-Native Mode for Trunk BCP
Cisco IOS Release 12.2(18)SXF	Support for the following SIP hardware was introduced on the Cisco 7600 series router and Catalyst 6500 series switch: • Cisco 7600 SIP-600 Support for the following features was introduced on the Cisco 7600 SIP-400: • Policing by committed information rate (CIR) percentage • QoS matching on class of service (CoS)—2-Port Gigabit Ethernet SPA only
Cisco IOS Release 12.2(18)SXE2	Support for the following SPA services card (SSC) was introduced on the Cisco 7600 series router and Catalyst 6500 series switch: • Cisco 7600 SSC-400
Cisco IOS Release 12.2(18)SXE	Support for the following SPA interface processor (SIP) hardware was introduced on the Cisco 7600 series router and Catalyst 6500 series switch: • Cisco 7600 SIP-200 • Cisco 7600 SIP-400

Supported SIP Features

The Cisco 7600 SIP-200, Cisco 7600 SIP-400, and Cisco 7600 SIP-600 are high-performance, feature-rich SPA interface processors that function as carrier cards for shared port adapters (SPAs) on the Cisco 7600 series router. These SIPs are supported on the Cisco 7600 series router and Catalyst 6500 series switch, and are compatible with one or more platform-independent SPAs. For more information on SPA compatibility, see the "SIP, SSC, and SPA Compatibility" section on page 2-3.

The Cisco 7600 series router is an edge aggregation router, and the SIPs provide a cost-effective solution for customers seeking moderate- to high-port density and line rate services:

- The Cisco 7600 SIP-200 provides WAN edge aggregation through lower-speed and low-density SPAs for network environments requiring regional office connectivity to headquarters, or collapsed LAN/WAN deployment.
- The Cisco 7600 SIP-400 provides higher-speed, high-density link aggregation for network environments requiring leased line and metro aggregation.
- The Cisco 7600 SIP-600 provides a high-speed interface for WANs and metro aggregation.

This section provides a list of some of the primary features supported by the SIP hardware and software. For feature compatibility information by SIP and SPA combination, and information about configuring these features, see Chapter 4, "Configuring the SIPs and SSC."

Cisco 7600 SIP-200 Features

- Field-programmable device (FPD) upgrade support

  The Cisco 7600 SIP-200 supports the standard FPD upgrade methods for the Cisco 7600 series router. For more information about FPD support, see Chapter 31, "Upgrading Field-Programmable Devices."

Cisco 7600 SIP-200 High Availability Features

- Automatic protection switching (APS)—ATM and POS SPAs
- Online insertion and removal (OIR) of the SIP and SPAs
- Nonstop Forwarding (NSF)
- Stateful switchover (SSO)—Not supported with dMLFR feature (dMLFR only supports RPR+)

Cisco 7600 SIP-200 ATM Features

- Aggregate Weighted Random Early Detection (WRED)
- ATM Adaptation Layer 5 (AAL5) Subnetwork Access Protocol (SNAP)
- AAL5 over Multiprotocol Label Switching (MPLS)
- ATM virtual circuit (VC) bundles
- RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5,* Multipoint Bridging (MPB) on the 2-Port and 4-Port OC-3c/STM-1 ATM SPA
- VC bundle Class of Service (CoS) precedence mapping

For a comprehensive list of supported and unsupported ATM features, SIP-dependent features, and restrictions see Chapter 6, "Overview of the ATM SPAs."

Cisco 7600 SIP-200 Frame Relay Features

For additional Frame Relay features, see also the MPLS and Quality of Service (QoS) feature sections.

Note Based on your link configuration, Multilink PPP (MLPPP) and Multilink Frame Relay (MLFR) are either software-based on the Cisco 7600 SIP-200, or hardware-based on the 8-Port Channelized T1/E1 SPA, 2-Port and 4-Port Channelized T3 SPA, and 1-Port Channelized OC-3/STM-1 SPA. For more information, see the corresponding configuration chapters for the SIPs and the serial SPAs.

- Distributed Multilink Frame Relay (dMLFR) (FRF.16)
- Distributed Link Fragmentation and Interleaving (dLFI) over Multilink PPP (MLPPP)
- dLFI with FRF.12
- Frame Relay over MPLS (FRoMPLS)
- Frame Relay VC bundles
- Frame Relay switching
- RFC 1490, *Multiprotocol Interconnect over Frame Relay*, Multipoint Bridging (MPB) on the 2-Port and 4-Port Clear Channel T3/E3 SPA, 2-Port and 4-Port Channelized T3 SPA, and the 8-Port Channelized T1/E1 SPA
- VC bundle Class of Service (CoS) precedence mapping

Cisco 7600 SIP-200 MPLS Features

- Explicit null
- Label disposition
- Label imposition
- Label swapping
- QoS tunneling
- Virtual private network (VPN) routing and forwarding (VRF) instance description
- dMLPPP with MPLS on VPN—Supported between the customer edge (CE) and provider edge (PE) devices
- Any Transport over MPLS (AToM) support, including:
  - ATM over MPLS (ATMoMPLS)—AAL5 VC mode
  - Ethernet over MPLS (EoMPLS)—(Single cell relay) VC mode
  - Frame Relay over MPLS (FRoMPLS)
  - FRoMPLS with dMLFR—Supported between the CE and PE devices
  - High-Level Data Link Control (HDLC) over MPLS (HDLCoMPLS)
  - PPP over MPLS (PPPoMPLS)—Not supported with dMLPPP or dLFI
- Hierarchical QoS for EoMPLS VCs

Beginning in Cisco IOS Release 12.2(33)SRA, the Cisco 7600 SIP-200 adds the following MPLS feature support:

- MPLS over RBE—ATM SPAs only

Beginning in Cisco IOS Release 12.2(33)SRB, the Cisco 7600 SIP-200 adds the following support:

- Software-based MLP bundles from 256 to 1024

Cisco 7600 SIP-200 MPLS Classification

- Default copy of IP precedence to MPLS experimental (EXP) bit
- Match on MPLS EXP bit using Modular QoS CLI (MQC)

Cisco 7600 SIP-200 MPLS Congestion Management

- Low latency queueing (LLQ)
- Class-based weighted fair queueing (CBWFQ)

Cisco 7600 SIP-200 MPLS Encapsulations

- ATM AAL5 SNAP
- Frame Relay
- HDLC
- MLPPP
- PPP

Cisco 7600 SIP-200 MPLS Marking

- Set MPLS EXP bit using MQC

Cisco 7600 SIP-200 MPLS Traffic Shaping

- Traffic shaping using MQC

Cisco 7600 SIP-200 Multiservice Features

- Compressed Real-Time Protocol (CRTP)
- FRF.11—Supported only in Cisco IOS Release 12.2(18)SXE and Cisco IOS Release 12.2(18)SXE2; Support for this feature was removed in Cisco IOS Release 12.2(18)SXF

Cisco 7600 SIP-200 QoS Features

This section provides a list of the Quality of Service (QoS) features that are supported by the Cisco 7600 SIP-200.

Cisco 7600 SIP-200 ATM SPA QoS Implementation

For the 2-Port and 4-Port OC-3c/STM-1 ATM SPA, the following applies:

- In the ingress direction, all Quality of Service (QoS) features are supported by the Cisco 7600 SIP-200.
- In the egress direction:
  - All queueing based features (such as class-based weighted fair queueing [CBWFQ], and ATM per-VC WFQ) are implemented on the Segmentation and Reassembly (SAR) processor on the SPA.
  - Policing is implemented on the SIP.
  - Class queue shaping is not supported.

Cisco 7600 SIP-200 Packet Marking

- IP precedence
- Differentiated Services Code Point (DSCP)
- Class-based marking
- ATM cell loss priority (CLP) to EXP marking/Type of Service (ToS)/DSCP
- Frame relay discard eligibility (DE) to EXP marking/ToS/DSCP

Cisco 7600 SIP-200 Policing and Dropping

- Aggregate
- Dual rate
- Hierarchical
- DSCP Markdown
- Policing—Precedence, DSCP marking
- Policing—EXP marking
- Explicit Drop in Class
- Matching packet length

Cisco 7600 SIP-200 Classification Into a Queue

- MPLS EXP
- ACL number
- Configurable queue size
- Network-based application recognition (NBAR)/dSTILE

Cisco 7600 SIP-200 Congestion Management

- Weighted fair queueing (WFQ)
- Class-based weighted fair queueing (CBWFQ)
- Per-VC CBWFQ
- Allocation, DSCP, EXP and precedence matching
- LLQ or priority queueing (strict priority only)
- Configurable LLQ burst size

Cisco 7600 SIP-200 Congestion Avoidance

- Random early detection (RED)
- Weighted random early detection (WRED)
- DiffServ-compliant WRED
- Aggregate WRED—ATM SPAs only

Cisco 7600 SIP-200 Shaping

- Generic traffic shaping (GTS)/Distributed traffic shaping (DTS)
- Hierarchical service policy with GTS
- Hierarchical traffic shaping with Frame Relay (FR)
- Hierarchical traffic shaping FR adaptive to FECN, BECN (Cisco 7600 SIP-200 only)
- Hierarchical traffic shaping for PPP and HDLC
- Ingress shaping
- Egress shaping

Note Egress shaping is not supported on the Cisco 7600 SIP-200 for the 2-Port and 4-Port OC-3c/STM-1 ATM SPA.

- Shaping by percentage

Cisco 7600 SIP-200 Other QoS Features

- Hierarchical QoS for EoMPLS VCs
- QoS with MLPPP

Beginning in Cisco IOS Release 12.2(33)SRA, the Cisco 7600 SIP-200 adds the following QoS feature support:

- QoS Support on bridging features

Cisco 7600 SIP-200 Fragmentation Features

- dLFI over ATM—Not supported with HA features
- dLFI over MLPPP
- dLFI with MPLS on VPN—Supported between the CE and PE devices
- FRF.12

Cisco 7600 SIP-200 Layer 2 Protocols and Encapsulation

- AAL5 Network Layer Protocol ID (NLPID)
- AAL5 SNAP
- Cisco Frame Relay
- IETF Frame Relay
- Frame Relay two-octet header
- Frame Relay BECN/FECN
- Frame Relay PVC
- Frame Relay UNI
- HDLC
- MLPPP
- PPP

Cisco 7600 SIP-200 Layer 2 Interworking

- ATM VC trunk emulation
- Bridged and routed RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*
- RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5,* Multipoint Bridging (MPB)
- RFC 1490, *Multiprotocol Interconnect over Frame Relay*, Multipoint Bridging (MPB)
- Bridging of Routed Encapsulations (BRE)
- Routed bridged encapsulation (RBE)

Note RBE is not supported when using the Intermediate System-to-Intermediate System (IS-IS) routing protocol.

- RFC 3518, *Point-to-Point Protocol (PPP) Bridging Control Protocol (BCP)*

Beginning in Cisco IOS Release 12.2(33)SRA, the Cisco 7600 SIP-200 adds the following Layer 2 interworking feature support:

- BCP support on 8-Port Channelized T1/E1 SPA, 2-Port and 4-Port Channelized T3 SPAs, 1-Port Channelized OC-3/STM-1 SPA, 2-Port and 4-Port Clear Channel T3/E3 SPAs, and 2-Port and 4-Port OC-3c/STM-1 POS SPAs
- BCP (trunk mode) support over MLPPP on 8-Port Channelized T1/E1 SPA, 2-Port and 4-Port Channelized T3 SPAs, and 1-Port Channelized OC-3/STM-1 SPA
- Multi-VC to VLAN scalability

- QoS support on bridging
- Software-based MLPPP
- Software-based MLFR

Cisco 7600 SIP-400 Features

- FPD upgrade support—The Cisco 7600 SIP-400 supports the standard FPD upgrade methods for the Cisco 7600 series router. For more information about FPD support, see Chapter 31, "Upgrading Field-Programmable Devices."
- Lawful Intercept—The Cisco 7600 SIP-400 supports Lawful Intercept in Cisco IOS Release 12.2(33)SRB and later.

Cisco 7600 SIP-400 High Availability Features

- Automatic protection switching (APS)—ATM and POS SPAs
- Online insertion and removal (OIR) of the SIP and SPAs
- Stateful switchover (SSO)

Cisco 7600 SIP-400 MPLS Features

Note For the Cisco 7600 SIP-400, the following MPLS features are implemented on the Supervisor Engine 720 (PFC3B and PFC3BXL) and the Route Switch Processor 720 (PFC3C and PFC3CXL): Label imposition, label swapping, label disposition, explicit null, default copy of IP precedence to EXP bit classification, and QoS tunneling. For more information about the requirements for Policy Feature Cards (PFCs) on the Cisco 7600 series router, refer to the *Release Notes for Cisco IOS Release 12.2SX on the Supervisor Engine 720, Supervisor Engine 32, and Supervisor Engine 2* at the following URL: http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/122sx/ol_4164.htm#wp2561312

- VRF description
- Any Transport over MPLS (AToM) support, including:
  - ATMoMPLS—AAL0 mode (single cell relay only)
  - ATMoMPLS—AAL5 mode
  - EoMPLS—Port mode
  - EoMPLS—VLAN mode
  - FRoMPLS—DLCI mode

Beginning in Cisco IOS Release 12.2(33)SRA, the Cisco 7600 SIP-400 adds the following MPLS feature support:

- Ethernet over MPLS (EoMPLS) VC scaling
- Ingress/Egress CoS classification with ingress policing per VLAN or EoMPLS VC
- Hierarchical VPLS (H-VPLS) with MPLS Edge
- Hierarchical QoS support for Ethernet over MPLS (EoMPLS) VCs

Cisco 7600 SIP-400 MPLS Congestion Management

- LLQ
- CBWFQ

Cisco 7600 SIP-400 MPLS Encapsulations

- ATM AAL5 SNAP
- Ethernet with 802.1q
- Frame Relay
- HDLC
- Generic Routing Encapsulation (GRE)—2-Port Gigabit Ethernet SPA only
- PPP

Cisco 7600 SIP-400 MPLS Marking

- Set MPLS EXP bits at tag imposition using MQC (**set mpls-experiment** command)—Input IP interface
- Set MPLS EXP bits on topmost label (set EXP topmost) using MQC (**set mpls-experiment topmost** command)—Input and output MPLS interface
- Mapping Ethernet 802.1q priority bits to MPLS EXP bits for EoMPLS

Cisco 7600 SIP-400 QoS Features

This section provides a list of the Quality of Service (QoS) features that are supported by the Cisco 7600 SIP-400.

Cisco 7600 SIP-400 Packet Marking

- IP precedence (**set ip precedence** command)—Input and output
- DSCP (**set dscp** command)—Input and output
- Class-based marking
- DE to EXP marking/ToS/DSCP
- CLP to EXP marking/ToS/DSCP
- Ethernet 802.1q priority bits to EXP marking (EoMPLS)

Cisco 7600 SIP-400 Policing and Dropping

- Dual rate
- Hierarchical
- Dual-rate policer with three-color marker
- Policing—Percent
- Policing—Precedence, DSCP marking
- Policing—EXP marking
- Policing—Set ATM CLP, FR DE

- Policing—Set MPLS EXP bits on topmost label (set EXP topmost)
- Explicit Drop in Class

Cisco 7600 SIP-400 Classification Into a Queue

- Access control lists (IPv4 and IPv6)
  - Access group (**match access-group** command)—Input and output
  - Address (IPv6 compress mode only)
  - Name
  - Number
  - Source and destination port
  - TCP flag (IPv4 only)
- ATM CLP (**match atm clp** command)—Input ATM interface
- Configurable queue size
- CoS (**match cos** command)—Input and output dot1q tagged frames for 2-Port Gigabit Ethernet SPA only
- Frame Relay DE (**match fr-de** command)—Input Frame Relay interface
- Inner CoS (**match cos inner** command)
- IP DSCP (**match dscp** command)—Input and output
- IP precedence (**match ip precedence** command)—Input and output
- MPLS EXP (**match mpls experimental** command)—Input and output MPLS interface
- Multiple matches per class map (up to 8)

Beginning in Cisco IOS Release 12.2(33)SRA, the Cisco 7600 SIP-400 adds the following QoS classification feature support:

- Ingress/Egress CoS classification with ingress policing per VLAN or EoMPLS VC

Cisco 7600 SIP-400 Congestion Management

- CBWFQ
- Per-VC CBWFQ
- DSCP, EXP and Precedence matching
- LLQ or priority queueing (strict priority only)
- Dual-priority queuing

Cisco 7600 SIP-400 Congestion Avoidance

- RED
- WRED
- DiffServ-compliant WRED
- Aggregate WRED—ATM SPAs only

Cisco 7600 SIP-400 Shaping

- Hierarchical traffic shaping using class-default (not supported for user-defined class)
- Hierarchical traffic shaping FR
- Hierarchical traffic shaping for PPP and HDLC
- Egress shaping

Cisco 7600 SIP-400 Fragmentation Features

- dLFI with ATM

Cisco 7600 SIP-400 Layer 2 Protocols and Encapsulation

- PPP
- AAL5 SNAP
- HDLC
- Cisco Frame Relay
- IETF Frame Relay
- Frame Relay two-octet header
- Frame Relay BECN/FECN
- Frame Relay PVC
- Frame Relay UNI

Cisco 7600 SIP-400 Layer 2 Interworking

- Bridged and routed RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*
- RFC 3518, *Point-to-Point Protocol (PPP) Bridging Control Protocol (BCP)*, on the 2-Port and 4-Port OC-3c/STM-1 POS SPA and 1-Port OC-12c/STM-4 POS SPA.

Beginning in Cisco IOS Release 12.2SRB1, the Cisco 7600 SIP-400 supports:

- Backup Interface for Flexible UNI (for Gigabit Ethernet SPAs)

Beginning in Cisco IOS Release 12.2(33)SRA, the Cisco 7600 SIP-400 supports:

- BCP on POS SPAs (OC-3c/STM-1, OC-12c/STM-4, OC-48c/STM-16, and OC-192c/STM-64)
- Multipoint Bridging (MPB)
- Multi-VC to VLAN scalability
- QoS support on bridging features

Cisco 7600 SIP-600 Features

- FPD upgrade support—The Cisco 7600 SIP-600 supports the standard FPD upgrade methods for the Cisco 7600 series router. For more information about FPD support, see Chapter 31, "Upgrading Field-Programmable Devices."
- Layer 2 switch port

- EtherChannel and Link Aggregate Control Protocol (IEEE 802.3ad)
- Control Plane Policing (CoPP)

Cisco 7600 SIP-600 High Availability Features

- Automatic protection switching (APS)
- Online insertion and removal (OIR) of the SIP and SPAs
- Nonstop Forwarding (NSF)
- Stateful switchover (SSO)

Cisco 7600 SIP-600 MPLS Features

- Unicast switching, with specific support for up to six label push operations, one label pop operation (two label pop operations in case of Explicit Null), or one label swap with up to five label push operations, at each MPLS switch node
- Support for Explicit Null label to preserve CoS information when forwarding packets from provider (P) to provider edge (PE) routers
- Support for Implicit Null label to request that penultimate hop router forward IP packets without labels to the router at the end of the label switch path (LSP)
- VRF
- Traffic engineering
- Any Transport over MPLS (AToM) support—EoMPLS only, including:
  - PFC-based (No MAC address learning)
  - SIP-based (MAC address learning, requires SIP as uplink)
  - Up to 4000 EoMPLS VCs per system
- Virtual Private LAN Service (VPLS) support, including:
  - H-VPLS with MPLS edge—H-VPLS with MPLS edge requires either an OSM or Cisco 7600 SIP-600 in both the downlink (facing UPE) and uplink (MPLS core). For more information about configuring H-VPLS, see Chapter 12, "Configuring the Fast Ethernet and Gigabit Ethernet SPAs."
  - H-VPLS with Q-in-Q edge—Requires Cisco 7600 SIP-600 in the uplink, and any LAN port or Cisco 7600 SIP-600 on the downlink
  - Up to 4000 VPLS domains
  - Up to 60 VPLS peers per domain
  - Up to 30,000 pseudowires, used in any combination of domains and peers up to the 4000-domain or 60-peer maximums; for example, support of up to 4000 domains with 7 peers or up to 60 peers in 500 domains
- MPLS Operation, Administration, and Maintenance (OAM) support, including:
  - LSP ping and traceroute
  - Virtual Circuit Connection Verification (VCCV)

Cisco 7600 SIP-600 Layer 2 Protocols and Encapsulation

- HDLC (Cisco Systems)
- PPP
- PPP over SONET/SDH
- Layer 2 Gigabit Ethernet support, including:
  - IEEE 802.3z 1000 Mbps Gigabit Ethernet
  - IEEE 802.3ab 1000BaseT Gigabit Ethernet
  - IEEE 802.3ae 10 Gbps Ethernet (1-Port 10-Gigabit Ethernet SPA only)
  - Jumbo frame (up to 9216 bytes)
  - ARPA, IEEE 802.3 SAP, IEEE 802.3 SNAP, Q-in-Q
  - IEEE 802.1q VLANs
  - Autonegotiation support including IEEE 802.3 flow control and pause frames
  - Gigabit Ethernet Channel (GEC)
  - IEEE 802.3ad link aggregation
  - Address Resolution Protocol (ARP)/Reverse ARP (RARP)
  - Hot Standby Router Protocol (HSRP)
  - Virtual Router Redundancy Protocol (VRRP)

Cisco 7600 SIP-600 QoS Features

This section provides a list of the Quality of Service (QoS) features that are supported by the Cisco 7600 SIP-600.

- MQC

Cisco 7600 SIP-600 Packet Marking

- IP precedence (**set ip precedence** command)
- DSCP (**set dscp** command)
- MPLS EXP (**match mpls experimental** command)

Note Mapping 802.1p CoS values to MPLS EXP bits is supported using EoMPLS only.

Cisco 7600 SIP-600 Policing and Dropping

- Input policing on a per-port and per-VLAN basis

Cisco 7600 SIP-600 Classification Into a Queue

- Input and output ACLs on a per-port and per-VLAN basis
- Input VLAN (**match input vlan** command)
- IP DSCP (**match dscp** command)

- IP precedence (**match ip precedence** command)
- MPLS EXP (**match mpls experimental** command)
- QoS group (**match qos-group** command)
- VLAN (**match vlan** command)

Cisco 7600 SIP-600 Congestion Management

- CBWFQ
- LLQ

Cisco 7600 SIP-600 Congestion Avoidance

- WRED

Cisco 7600 SIP-600 Shaping

- Output shaping on a per-port and per-VLAN basis
- Output hierarchical traffic shaping—Two levels of shaping on an interface, subinterface, or group of subinterfaces

Supported SSC Features

The Cisco 7600 SSC-400 is a streamlined services card that provides a very high bandwidth data path between the Cisco 7600 series router platform backplane and the high-speed interconnects on the IPSec VPN SPA.

For more information about the features and configuration supported by the IPSec VPN SPA with the Cisco 7600 SSC-400, see the related chapters in the IPSec VPN Shared Port Adapter part of this book.

Cisco 7600 SSC-400 Features

- Support of up to two IPSec VPN SPAs per slot
- Online insertion and removal (OIR) of the SSC and SPAs

Restrictions

This section documents unsupported features and feature restrictions for the SIPs and SSC on the Cisco 7600 series router.

Cisco 7600 SIP-200 Restrictions

As of Cisco IOS Release 12.2(18)SXE, the Cisco 7600 SIP-200 has the following restrictions:

- The Cisco 7600 SIP-200 is not supported with a Supervisor Engine 1, Supervisor Engine 1A, Supervisor Engine 2, or Supervisor Engine 720A.

- A maximum number of 200 PVCs or SVCs using Link Fragmentation and Interleaving (LFI) is supported for all ATM SPAs (or other ATM modules) in a Cisco 7600 series router.
- The following features are not supported:
  - ATM LAN Emulation (LANE)
  - dLFI over Frame Relay (dLFIoFR)
  - dLFI with MPLS
  - Layer 2 Tunneling Protocol (L2TP) version 2
  - L2TP version 3
  - Legacy Priority Queueing and Custom Queueing
  - PPP over Ethernet (PPPoE)
  - Reliable PPP (RFC 1663, *PPP Reliable Transmission*)
  - Stacker Compression (STAC)
  - X.25, Link Access Procedure, Balanced (LAPB)
- PPP over MPLS (PPPoMPLS) is not supported with dMLPPP or dLFI.
- High availability (HA) features have some restrictions when configured with the following distributed features on the Cisco 7600 SIP-200:
  - When you configure HA with dMLFR, the Cisco 7600 SIP-200 only supports RPR+.
  - HA features with dLFI over ATM (dLFIoATM) are not supported.
  - HA features with dLFI over Frame Relay (dLFIoFR) are not supported.

Cisco 7600 SIP-400 Restrictions

In Cisco IOS Release 12.2(18)SXE and later, the Cisco 7600 SIP-400 has the following restrictions:

- The Cisco 7600 SIP-400 is not supported with a Supervisor Engine 1, Supervisor Engine 1A, or Supervisor Engine 2. It is also not supported with a Supervisor Engine 720 PFC3A, or in PFC3A mode.

  For more information about the requirements for Policy Feature Cards (PFCs) on the Cisco 7600 series router, refer to the *Release Notes for Cisco IOS Release 12.2SX on the Supervisor Engine 720, Supervisor Engine 32, and Supervisor Engine 2* at the following URL:
  http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/122sx/ol_4164.htm#wp2561312
- The Cisco 7600 SIP-400 is not supported with PFC-2 based systems.
- A maximum number of 200 PVCs or SVCs using Link Fragmentation and Interleaving (LFI) is supported for all ATM SPAs (or other ATM modules) in a Cisco 7600 series router.
- For AToM in Cisco IOS 12.2SX releases, the Cisco 7600 SIP-400 does not support the following features when they are located in the data path. This means you should not configure the following features if the SIP is facing the customer edge (CE) or the MPLS core:
  - HDLCoMPLS
  - PPPoMPLS
  - Virtual Private LAN Service (VPLS)
- For AToM beginning in Cisco IOS Release 12.2(33)SRA, the Cisco 7600 SIP-400 supports the following features on CE-facing interfaces:

  - HDLCoMPLS
  - PPPoMPLS
  - VPLS
- The Cisco 7600 SIP-400 supports EoMPLS with directly connected provider edge (PE) devices when the Cisco 7600 SIP-400 is on the MPLS core side of the network.
- The Cisco 7600 SIP-400 does not support the ability to enable or disable tunneling of Layer 2 packets, such as for the VLAN Trunking Protocol (VTP), Cisco Discovery Protocol (CDP), and bridge protocol data unit (BPDU). The Cisco 7600 SIP-400 tunnels BPDUs, and always blocks VTP and CDP packets from the tunnel.
- In ATMoMPLS AAL5 and cell mode, the Cisco 7600 SIP-400 supports non-matching VPIs/VCIs between PEs if the Cisco 7600 SIP-400 is on both sides of the network.
- The Cisco 7600 SIP-400 supports matching on FR-DE to set MPLS-EXP for FRoMPLS.
- The Cisco 7600 SIP-400 supports use of the **xconnect** command to configure AToM circuits for all AToM connection types except ATMoMPLS. For ATMoMPLS, you must use the **mpls l2 transport route** command.
- The Cisco 7600 SIP-400 does not support the following QoS classification features with AToM:
  - Matching on data-link connection identifier (DLCI) is unsupported.
  - Matching on virtual LAN (VLAN) is unsupported.
  - Matching on class of service (CoS) is unsupported in Cisco IOS Release 12.2(18)SXE and Cisco IOS Release 12.2(18)SXE2 only. Beginning in Cisco IOS Release 12.2(18)SXF, it is supported with the 2-Port Gigabit Ethernet SPA.
  - Matching on input interface is unsupported.
  - Matching on packet length is unsupported.
  - Matching on media access control (MAC) address is unsupported.
  - Matching on protocol type, including Border Gateway Protocol (BGP), is unsupported.
- The Cisco 7600 SIP-400 does not support the following QoS classification features using MQC:
  - ACL IPv6 full address
  - ACL IPv6 TCP flags
  - Class map (**match class-map** command)
  - CoS inner (**match cos inner** command)—Supported beginning in Cisco IOS Release 12.2(33)SRA on 2-Port Gigabit Ethernet SPA input and output interfaces and with bridging features.
  - Destination sensitive services (DSS)
  - Discard class (**match discard-class** command)
  - Frame Relay DLCI (**match fr-dlci** command)—Supported beginning in Cisco IOS Release 12.2(33)SRA on Frame Relay input and output interfaces and with Frame Relay bridging features.
  - Input interface (**match input-interface** command)
  - Input VLAN (**match input vlan** command)—Supported beginning in Cisco IOS Release 12.2(33)SRA on output interfaces only.
  - IP RTP (**match ip rtp** command)
  - IPv4 and IPv6 ToS

- MAC address (**match mac** command)
- Match protocol (**match protocol** command)—Supports IP only.
- Packet length (**match packet length** command)
- QoS group (**match qos-group** command)
- Source and destination autonomous system (AS) (**match as** command)
- Source and destination Border Gateway Protocol (BGP) community (**match bgp-community** command)
- VLAN (**match vlan** command)
- VLAN inner (**match vlan inner** command)—Supported beginning in Cisco IOS Release 12.2(33)SRA on input and output interfaces and with bridging features.

- The Cisco 7600 SIP-400 does not support the following QoS marking features:
  - CoS (**set cos** command)
  - CoS inner (**set cos inner** command)
- The Cisco 7600 SIP-400 does not support the following QoS marking features using MQC:
  - QoS group (**set qos-group** command)
  - Next-hop (**set next-hop** command)
  - Discard class (**set discard-class** command)
  - Table (**set table** command)
- The Cisco 7600 SIP-400 does not support the following QoS queueing actions using MQC:
  - Flow-based queueing
  - Adaptive shaping
- The Cisco 7600 SIP-400 does not support the following QoS policing feature:
  - Policing by Committed Information Rate (CIR) percentage (**police cir percent** command)—Supported as of Cisco IOS Release 12.2(18)SXF.
- The Cisco 7600 SIP-400 does not support the following Frame Relay features:
  - Matching on DLCI is unsupported.
  - Bridging encapsulation is unsupported.
  - Multicast on multipoint interfaces is unsupported.
  - FRF.5 is unsupported.
  - FRF.8 is unsupported.
  - FRF.12 fragmentation is unsupported.
  - FRF.16 multilink support of four-octet extended addressing on an SVC is unsupported.
  - NNI is unsupported.
  - PVC bundling is unsupported.
  - PPP over Frame Relay is unsupported.
- The Cisco 7600 SIP-400 does not support RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5,* Multipoint Bridging (MPB). However, point-to-point bridging is supported.

- As of Cisco IOS Release 12.2(18)SXF, when using the Cisco 7600 SIP-400 with the 2-Port Gigabit Ethernet SPA or the 1-Port OC-48c/STM-16 ATM SPA, consider the following oversubscription guidelines:
  - The Cisco 7600 SIP-400 only supports installation of one 1-Port OC-48c/STM-16 ATM SPA without any other SPAs installed in the SIP.
  - The Cisco 7600 SIP-400 supports installation of up to two 2-Port Gigabit Ethernet SPAs without any other SPAs installed in the SIP.
  - The Cisco 7600 SIP-400 supports installation of any combination of OC-3 or OC-12 POS or ATM SPAs, up to a combined ingress bandwidth of OC-48 rates.
  - The Cisco 7600 SIP-400 supports installation of any combination of OC-3 or OC-12 POS or ATM SPAs up to a combined ingress bandwidth of OC-24 rates, when installed with a single 2-Port Gigabit Ethernet SPA.
- Q-in-Q (the ability to map a single 802.1Q tag or a random double tag combination into a VPLS instance, a Layer 3 MPLS VPN, or an EoMPLS VC) is not supported.
- Cisco Discovery Protocol (CDP) is disabled by default on the 2-Port Gigabit Ethernet SPA interfaces and subinterfaces on the Cisco 7600 SIP-400.

Cisco 7600 SIP-600 Restrictions

As of Cisco IOS Release 12.2(18)SXF, the Cisco 7600 SIP-600 has the following restrictions:

- The Cisco 7600 SIP-600 is not supported by the Supervisor Engine 32 or the Supervisor Engine 720 with PFC3A.

  For more information about the requirements for Policy Feature Cards (PFCs) on the Cisco 7600 series router, refer to the *Release Notes for Cisco IOS Release 12.2SX on the Supervisor Engine 720, Supervisor Engine 32, and Supervisor Engine 2* at the following URL:
  http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/122sx/ol_4164.htm#wp2561312
- The Cisco 7600 SIP-600 supports installation of only a single SPA in the first subslot.
- Removal of one type of SPA and reinsertion of a different type of SPA during OIR causes a reload of the Cisco 7600 SIP-600.
- Q-in-Q (the ability to map a single 802.1Q tag or a random double tag combination into a VPLS instance, a Layer 3 MPLS VPN, or an EoMPLS VC) is not supported.
- H-VPLS with MPLS edge requires either an OSM or Cisco 7600 SIP-600 in both the downlink (facing UPE) and uplink (MPLS core).
- Output policing is not supported.
- The aggregate guaranteed bandwidth configured for all QOS policies applied to a main interface cannot exceed the bandwidth of the link. 1% of the link rate bandwidth is reserved for control packet traffic. The remaining 99% of guaranteed rates are available for QoS configuration. For policies applied to the main interface, an attempt is made to acquire the 1% guaranteed rate from class-default. If control packet bandwidth can not be acquired, then errors are reported in the log file.
- On any Cisco 7600 SIP-600 Ethernet port subinterface using VLANs, a unique VLAN ID must be assigned. This VLAN ID cannot be in use by any other interface on the Cisco 7600 series router.

Cisco 7600 SSC-400 Restrictions

As of Cisco IOS Release 12.2(18)SXE2, the Cisco 7600 SSC-400 has the following restrictions:

- The Cisco 7600 SSC-400 is only supported by the Supervisor Engine 720 (MSFC3 and PFC3).

  For more information about the requirements for Policy Feature Cards (PFCs) on the Cisco 7600 series router, refer to the *Release Notes for Cisco IOS Release 12.2SX on the Supervisor Engine 720, Supervisor Engine 32, and Supervisor Engine 2* at the following URL:
  http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/122sx/ol_4164.htm#wp2561312
- The Cisco 7600 SSC-400 only supports two IPSec VPN SPAs.

As of Cisco IOS Release 12.2(18)SXF, the Cisco 7600 SSC-400 has the following restrictions:

- The Cisco 7600 SSC-400 is not supported by the Supervisor Engine 32. The Cisco 7600 SSC-400 is only supported by the Supervisor Engine 720 (MSFC3 and PFC3).

  For more information about the requirements for Policy Feature Cards (PFCs) on the Cisco 7600 series router, refer to the *Release Notes for Cisco IOS Release 12.2SX on the Supervisor Engine 720, Supervisor Engine 32, and Supervisor Engine 2* at the following URL:
  http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/122sx/ol_4164.htm#wp2561312
- The Cisco 7600 SSC-400 only supports two IPSec VPN SPAs.

Supported MIBs

The following MIBs are supported in Cisco IOS Release 12.2(18)SXE and later for the Cisco 7600 SIP-200 on a Cisco 7600 series router:

- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-EXT-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- ENTITY-MIB
- OLD-CISCO-CHASSIS-MIB

The following MIBs are supported in Cisco IOS Release 12.2(18)SXE and later for the Cisco 7600 SIP-400 on a Cisco 7600 series router:

- ATM-ACCOUNTING-INFORMATION-MIB (RFC 2512)
- ATM-MIB (RFC 2515)
- ATM-SOFT-PVC-MIB
- ATM-TC-MIB
- ATM-TRACE-MIB
- CISCO-AAL5-MIB
- CISCO-ATM-CONN-MIB
- CISCO-ATM-RM-MIB
- CISCO-ATM TRAFFIC-MIB
- CISCO-CLASS-BASED-QOS-MIB
- CISCO-ENTITY-ASSET-MIB

- CISCO-ENTITY-EXT-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- ENTITY-MIB
- IF-MIB
- OLD-CISCO-CHASSIS-MIB
- SONET MIB (RFC 2558)

The following MIBs are supported in Cisco IOS Release 12.2(18)SXF and later for the Cisco 7600 SIP-600 on a Cisco 7600 series router:

- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-EXT-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- ENTITY-MIB
- OLD-CISCO-CHASSIS-MIB

The following MIBs are supported in Cisco IOS Release 12.2(18)SXE2 and later for the Cisco 7600 SSC-400 on a Cisco 7600 series router:

- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-EXT-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- ENTITY-MIB
- OLD-CISCO-CHASSIS-MIB

For more information about MIB support on a Cisco 7600 series router, refer to the *Cisco 7600 Series Internet Router MIB Specifications Guide*, at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/prod_technical_reference_list.html

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

Displaying the SIP and SSC Hardware Type

To verify the SIP or SSC hardware type that is installed in your Cisco 7600 series router, you can use the **show module** command. There are other commands on the Cisco 7600 series router that also provide SIP and SSC hardware information, such as the **show idprom** command and **show diagbus** command.

Table 3-1 shows the hardware description that appears in the **show module** and **show idprom** command output for each type of SIP that is supported on the Cisco 7600 series router.

Table 3-1 SIP Hardware Descriptions in show Commands

SIP	Description in show module and show idprom Commands
Cisco 7600 SIP-200	4-subslot SPA Interface Processor-200 / 7600-SIP-200
Cisco 7600 SIP-400	4-subslot SPA Interface Processor-400 / 7600-SIP-400
Cisco 7600 SIP-600	1-subslot SPA Interface Processor-600 / 7600-SIP-600
Cisco 7600 SSC-400	2-subslot Services SPA Carrier-400 / 7600-SSC-400

Example of the show module Command

The following example shows output from the **show module** command on the Cisco 7600 series router with a Cisco 7600 SIP-400 installed in slot 13:

```
Router# show module 13
Mod Ports Card Type                              Model              Serial No.
--- ----- -------------------------------------- ------------------ -----------

 13    0  4-subslot SPA Interface Processor-400  7600-SIP-400       JAB0851042X

Mod MAC addresses                       Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
 13  00e0.aabb.cc00 to 00e0.aabb.cc3f   0.525 12.2(PP_SPL_ 12.2(PP_SPL_ Ok

Mod Online Diag Status
--- -------------------
 13 Pass
```

Example of the show idprom Command

The following example shows sample output for a Cisco 7600 SIP-200 installed in slot 4 of the router:

```
Router# show idprom module 4
IDPROM for module #4
  (FRU is '4-subslot SPA Interface Processor-200')
  OEM String = 'Cisco Systems'
  Product Number = '7600-SIP-200'
  Serial Number = 'SAD0738006Y'
  Manufacturing Assembly Number = '73-8272-03'
  Manufacturing Assembly Revision = '03'
  Hardware Revision = 0.333
  Current supplied (+) or consumed (-) = -4.77A
```

SIP-200 and SIP-400 Network Clock Distribution

The Cisco 7600 series routers have a distributed clocking system with two 8 KHZ backplane reference clocks that connect to every slot in the backplane to provide an egress (Tx) timing reference for the SPAs.

Starting with Cisco IOS release 12.2(33)SRB,the SIP-200 or SIP-400 can take clock input from various clock sources and distribute the clock to other supported cards by way of the chassis backplane to allow network operators to synchronize the transmit clocks of serial interfaces to a central timing reference. Synchronization to a central timing reference can help eliminate frame slips and associated loss of data on SONET and SDH interfaces.

Both the SIP-200 and the SIP-400 can act as the source that drives the backplane reference clocks by other SIPs. When a SIP-200 or SIP-400 is the source of the clocks, it uses the recovered Rx clock from any one of its SPAs input ports (see Table 3-2 for which SPAs support this functionality) to derive an 8 KHZ clock that it drives onto one or both backplane signals, or provides its own stratum 3 clock to the backplane.

Both the SIP-200 and the SIP-400 can also receive backplane clocks for use by their SPAs. When the SIP-200 and the SIP-400 receives backplane clocks, the clocks are dejittered and provided to the SPAs.

Table 3-2 shows reference clock sources. Table 3-3 shows the reference clock sources available for mapping to the backplane. Table 3-4 shows the clocks available to specific line cards.

Table 3-2 Reference Clock Sources

Reference Clock Input for Data Transmission	SIP-200	SIP-400
Local	All supported SONET/Serial SPAs	All supported SONET/Serial SPAs
Line	All supported SONET/Serial SPAs	All supported SONET/Serial SPAs
BITS Input	SPA-8XCHT1/E1	SPA-24CHT1-CE-ATM

Table 3-3 Reference Clock Sources Available for Mapping to Backplane

Clock Source	Line Card	SPA	Clock Derived From
Internal Oscillator	SIP-200	Not applicable	Not applicable
	SIP-400	Not applicable	Not applicable
Interface	SIP-200		SONET/SDH
		SPA-2XOC3-POS, SPA-4XOC3-POS	
		SPA-2XOC3-ATM, SPA-4XOC3-ATM	
	SIP-400	SPA-1CHOC3-CE-ATM	
		SPA-2XOC3-POS, SPA-4XOC3-POS	
		SPA-1XOC12-POS	
		SPA-1XOC48-POS	
		SPA-2XOC3-ATM, SPA-4XOC3-ATM	
		SPA-1XOC12-ATM	
		SPA-1XOC-48ATM	

Table 3-3 Reference Clock Sources Available for Mapping to Backplane

Clock Source	Line Card	SPA	Clock Derived From
Controller	SIP-200	SPA-8XCHT1/E1	T1/E1
		SPA-1XCHSTM1/OC3	STM1/OC3
		SPA-2XT3/E3, SPA-4XT3/E3	Cannot provide clock to backplane
		SPA-2XCT3/DS0, SPA-4XCT3/DS0	Cannot provide the clock to backplane

Table 3-4 Line Cards Able to Receive Clocks from Backplane

Line Card	SPA	Minimum Interface Level for Clock Source Input
SIP-200	SPA-8XCHT1/E1	Cannot take clock from backplane
	SPA-2XT3/E3, SPA-4XT3/E3	Cannot take clock from backplane
	SPA-2XCT3/DS0, SPA-4XCT3/DS0	Cannot take clock from backplane
	SPA-1XCHSTM1/OC3	STM1/OC3
	SPA-2XOC3-POS, SPA-4XOC3-POS	
	SPA-2XOC3-ATM, SPA-4XOC3-ATM	
SIP-400	SPA-24CHT1-CE-ATM	T1/E1
	SPA-1CHOC3-CE-ATM	STM1/OC3
	SPA-2XOC3-POS, SPA-4XOC3-POS	
	SPA-1XOC12-POS	STM4/OC12
	SPA-2XOC3-ATM, SPA-4XOC3-ATM	STM1/OC3
	SPA-1XOC12-ATM	STM4/OC12
	SPA-1XOC-48ATM	STM16/OC48

For additional information, see BITS Clock Support—Receive and Distribute—CEoP SPA on SIP-400, page 10-28.

CHAPTER 4

Configuring the SIPs and SSC

This chapter provides information about configuring SIPs and SSCs on the Cisco 7600 series router. It includes the following sections:

- Configuration Tasks, page 4-1
- Configuration Examples, page 4-106

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications that correspond to your Cisco IOS software release.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Configuration Tasks

This section describes how to configure the SIPs and SSCs and includes information about verifying the configuration.

It includes the following topics:

- Required Configuration Tasks, page 4-2
- Identifying Slots and Subslots for SIPs, SSCs, and SPAs, page 4-2
- Configuring Compressed Real-Time Protocol, page 4-4
- Configuring Frame Relay Features, page 4-6
- Configuring Layer 2 Interworking Features on a SIP, page 4-20
- Configuring MPLS Features on a SIP, page 4-50
- Configuring QoS Features on a SIP, page 4-61
- Configuring Lawful Intercept on a Cisco 7600 SIP-400, page 4-87
- Configuring Security ACLs on an Access Interface on a Cisco 7600 SIP-400, page 4-88
- Configuring CoPP on the Cisco 7600 SIP-400, page 4-89
- Configuring IGMP Snooping on a SIP-200, page 4-93
- Configuring ACFC and PFC Support on Multilink Interfaces, page 4-94
- Configuring PPPoEoE on a Cisco 7600 SIP-400, page 4-97

- Configuring Source IPv4 and Source MAC Address Binding on the SIP-400, page 4-101
- Resetting a SIP, page 4-105

This section identifies those features that have SIP-specific configuration guidelines for you to consider and refers you to the supporting platform documentation.

Many of the Cisco IOS software features on the Cisco 7600 series router that the FlexWAN and Enhanced FlexWAN modules support, the SIPs also support. Use this chapter while also referencing the list of supported features on the SIPs in Chapter 3, "Overview of the SIPs and SSC."

Note When referring to the other platform documentation, be sure to note any SIP-specific configuration guidelines described in this document.

For information about configuring other features supported on the Cisco 7600 series router but not discussed in this document, refer to the *Cisco 7600 Series Cisco IOS Software Configuration Guide, 12.2SX* at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/products_configuration_guide_book09186a00801d4269.html

Required Configuration Tasks

As of Cisco IOS Release 12.2(18)SXE, there are not any features that require direct configuration on the SIP or SSC. This means that you do not need to attach to the SIP or SSC itself to perform any configuration.

However, the Cisco 7600 SIP-200 and Cisco 7600 SIP-400 do implement and support certain features that are configurable at the system level on the Route Processor (RP).

Identifying Slots and Subslots for SIPs, SSCs, and SPAs

This section describes how to specify the physical locations of a SIP and SPA on the Cisco 7600 series routers within the command-line interface (CLI) to configure or monitor those devices.

Note For simplicity, any reference to "SIP" in this section also applies to the SSC.

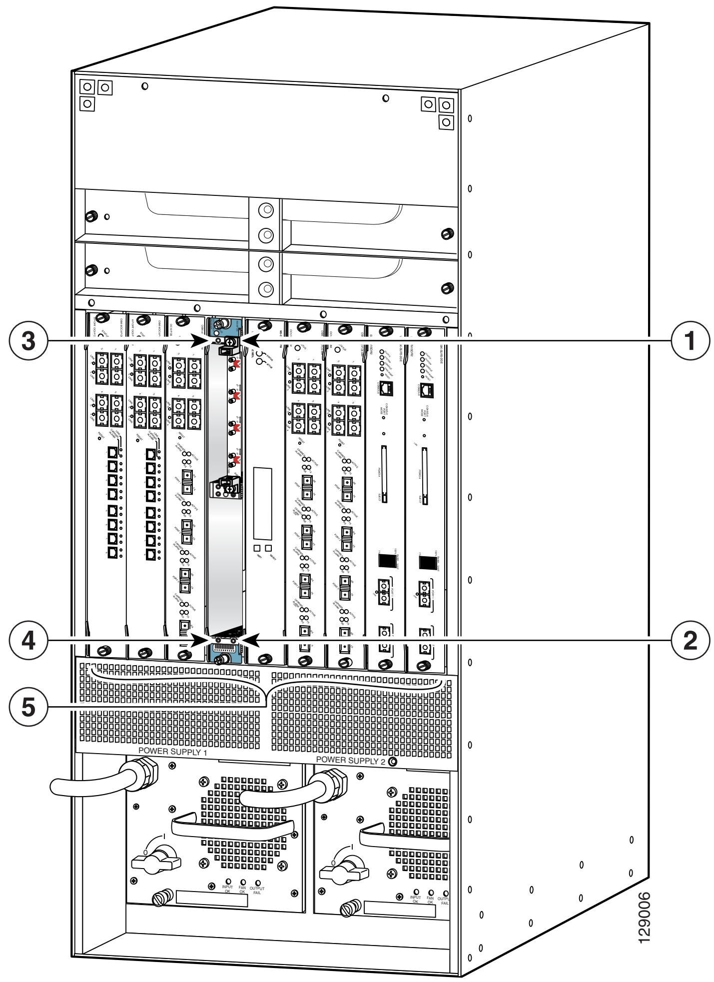
Specifying the Slot Location for a SIP or SSC

The Cisco 7600 series router supports different chassis models, each of which supports a certain number of chassis slots.

Note The Cisco 7600 series router SIPs are not supported with a Supervisor Engine 1, Supervisor Engine 1A, Supervisor Engine 2, or Supervisor Engine 720-3A.

Figure 4-1 shows an example of a SIP installed in slot 6 on a Cisco 7609 router. The Cisco 7609 router has nine vertically-oriented chassis slots, which are numbered 1 to 9 from right to left.

Figure 4-1 SIP and SPA Installed in a Cisco 7609 Router



1	SIP subslot 0	**4**	SIP subslot 3
2	SIP subslot 1	**5**	Chassis slots 1–9 (numbered from right to left)
3	SIP subslot 2		

Some commands allow you to display information about the SIP itself, such as **show module**, **show sip-disk**, **show idprom module**, **show hw-module slot**, and **show diagbus**. These commands require you to specify the chassis slot location where the SIP that you want information about is installed.

For example, to display status and information about the SIP installed in slot 6 as shown in Figure 4-1, enter the following command:

```
Router# show module 6
```

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*..

Specifying the SIP or SSC Subslot Location for a SPA

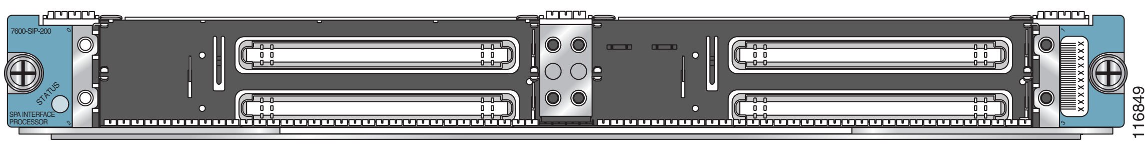
SIP subslots begin their numbering with “0” and have a horizontal or vertical orientation depending on the orientation of the SIP in the router chassis slot, as shown in the “SIP, SSC, and SPA Product Overview” chapter of the Cisco 7600 Series Router SIP, SSC, and SPA Software Configuration Guide.

Figure 4-1 shows an example of a Cisco 7600 SIP-200 installed with a vertical orientation on a Cisco 7609 router. The Cisco 7600 SIP-200 supports four subslots for the installation of SPAs. In this example, the subslot locations are vertically oriented as follows:

- SIP subslot 0—Top–right subslot
- SIP subslot 1—Bottom–right subslot
- SIP subslot 2—Top–left subslot
- SIP subslot 3—Bottom–left subslot

Figure 4-2 shows the faceplate for the Cisco 7600 SIP-200 in a horizontal orientation.

Figure 4-2 Cisco 7600 SIP-200 Faceplate



In this view, the subslot locations in a horizontal orientation are as follows:

- SIP subslot 0—Top–left subslot
- SIP subslot 1—Top–right subslot
- SIP subslot 2—Bottom–left subslot
- SIP subslot 3—Bottom–right subslot

The SIP subslot numbering is indicated by a small numeric label beside the subslot on the faceplate.

Just as with the SIPs, some commands allow you to display information about the SPA itself, such as **show idprom module** and **show hw-module subslot**. These commands require you to specify both the physical location of the SIP and SPA in the format, *slot*/*subslot*, where:

- *slot*—Specifies the chassis slot number in the Cisco 7600 series router where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.

For example, to display the operational status for the SPA installed in the first subslot of the SIP in chassis slot 6 shown in Figure 4-1, enter the following command:

```
Router# show hw-module subslot 6/0 oir
```

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*.

Configuring Compressed Real-Time Protocol

Compressed Real-Time Protocol (CRTP), from RFC 1889 (*RTP: A Transport Protocol for Real-Time Applications*), provides bandwidth efficiencies over low-speed links by compressing the UDP/RTP/IP header when transporting voice. With CRTP, the header for Voice over IP traffic can be reduced from

40 bytes to approximately 2 to 5 bytes offering substantial bandwidth efficiencies for low-speed links. CRTP is supported over Frame Relay, ATM, PPP, distributed MLPPP (dMLP PP), and HDLC encapsulated interfaces.

Table 4-1 provides information about where the CRTP feature for SPA interfaces is supported.

Table 4-1 CRTP Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Hardware-based CRTP	In Cisco IOS Release 12.2(18)SXE and later: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA	Not supported.	Not supported.
Hardware- and software-based CRTP	In Cisco IOS Release 12.2(33)SRA: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA • 1-Port Channelized OC-3/STM-1 SPA	Not supported.	Not supported.
CRTP with dLFIoLL—Only supported with one link present on the multilink interface	In Cisco IOS Release 12.2(18)SXE and later: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA Support for the following SPA was added in Cisco IOS Release 12.2(33)SRA: • 1-Port Channelized OC-3/STM-1 SPA	Not supported.	Not supported.
CRTP with dMLPPP	Supported. Not supported if LFI is enabled.	Not supported.	Not supported.
CRTP with dMLPPP and MPLS	Not supported.	Not supported.	Not supported.

CRTP Configuration Guidelines

To support CRTP on the Cisco 7600 SIP-200, consider the following guidelines:

- High-level Data Link Control (HDLC), PPP, or Frame Relay encapsulation must be configured.
- TCP or RTP header compression, or both, must be enabled.
- When distributed fast-switching is enabled, the **detail** option is not available with the **show ip rtp header-compression** and **show ip tcp header-compression** commands. Users who need the detailed information for either of these commands can retrieve this information by disabling distributed fast-switching and then entering the **show ip rtp header-compression detail** or **show ip tcp header-compression detail** commands.
- When using CRTP with distributed features on the Cisco 7600 SIP-200, consider the following guidelines and restrictions:
  - Hardware- and software-based CRTP is supported with Distributed Link Fragmentation and Interleaving over Leased Lines (dLFIoLL) if only one link is present on the multilink interface.
  - The following restrictions apply to Multilink PPP interfaces that use LFI:

    If RTP header compression is configured, RTP packets originating on or destined to the router will be fast-switched if the link is limited to one channel. If the link has more than one channel, the packets will be process-switched.

CRTP should not be configured on a multilink interface when LFI is enabled on the multilink interface if the multilink bundle has more than one member link, and a QoS policy with a feature is enabled on the multilink interface.

Note In a dMLPPP/dLFI configuration, packets do not carry the MLPPP header and sequence number. Thus, MLPPP distributes the packets across all member links. As a result, packets that are compressed by CRTP may arrive out-of-order at the receiving router. This prohibits CRTP from decompressing the packet header and forces CRTP to drop the packets.

For information on configuring CRTP, see *Configuring Distributed Compressed Real-Time Protocol* at the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a00800b75cd.html

Configuring Frame Relay Features

Many of the Frame Relay features supported on the FlexWAN and Enhanced FlexWAN modules on the Cisco 7600 series router are also supported by the SIPs. For a list of the supported Frame Relay features on the SIPs, see Chapter 3, "Overview of the SIPs and SSC."

This section describes those Frame Relay features that have SIP-specific configuration guidelines. After you review the SIP-specific guidelines described in this document, then refer to the referenced URLs for more information about configuring Frame Relay features.

The Frame Relay features for SIPs and SPAs are qualified as *distributed features* because the processing for the feature is handled by the SIP or SPA, or a combination of both.

Configuring Distributed Multilink Frame Relay (FRF.16) on the Cisco 7600 SIP-200

The Distributed Multilink Frame Relay (dMLFR) feature provides a cost-effective way to increase bandwidth for particular applications by enabling multiple serial links to be aggregated into a single bundle of bandwidth. Multilink Frame Relay is supported on the User-Network Interface (UNI) and the Network-to-Network Interface (NNI) in Frame Relay networks.

Note Based on your link configuration, dMLFR can be either software-based on the Cisco 7600 SIP-200, or hardware-based on the 8-Port Channelized T1/E1 SPA, 2-Port and 4-Port Channelized T3 SPAs, and 1-Port Channelized OC-3/STM-1 SPA. For more information about the hardware-based configuration, see also Chapter 17, "Configuring the 8-Port Channelized T1/E1 SPA," and Chapter 19, "Configuring the 2-Port and 4-Port Channelized T3 SPAs."

Table 4-2 provides information about where the dMLFR feature for SPA interfaces is supported.

Table 4-2 dMLFR Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Hardware- and software-based dMLFR	In Cisco IOS Release 12.2(18)SXE and later: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA In Cisco IOS Release 12.2(33)SRA and later: • 1-Port Channelized OC-3/STM-1 SPA InCisco IOS Release 12.2(33)SRC and later:	Not supported.	Not supported.

This section includes the following topics:

Overview of dMLFR

The Distributed Multilink Frame Relay feature enables you to create a virtual interface called a *bundle* or *bundle interface*. The bundle interface emulates a physical interface for the transport of frames. The Frame Relay data link runs on the bundle interface, and Frame Relay virtual circuits are built upon it.

The bundle is made up of multiple serial links, called *bundle links*. Each bundle link within a bundle corresponds to a physical interface. Bundle links are invisible to the Frame Relay data-link layer, so Frame Relay functionality cannot be configured on these interfaces. Regular Frame Relay functionality that you want to apply to these links must be configured on the bundle interface. Bundle links are visible to peer devices. The local router and peer devices exchange link integrity protocol control messages to determine which bundle links are operational and to synchronize which bundle links should be associated with which bundles.

For link management, each end of a bundle link follows the MLFR link integrity protocol and exchanges link control messages with its peer (the other end of the bundle link). To bring up a bundle link, both ends of the link must complete an exchange of ADD_LINK and ADD_LINK_ACK messages. To maintain the link, both ends periodically exchange HELLO and HELLO_ACK messages. This exchange of hello messages and acknowledgments serves as a keepalive mechanism for the link. If a router is sending hello messages but not receiving acknowledgments, it will resend the hello message up to a configured maximum number of times. If the router exhausts the maximum number of retries, the bundle link line protocol is considered down (unoperational).

The bundle link interface's line protocol status is considered up (operational) when the peer device acknowledges that it will use the same link for the bundle. The line protocol remains up when the peer device acknowledges the hello messages from the local router.

The bundle interface's line status becomes up when at least one bundle link has its line protocol status up. The bundle interface's line status goes down when the last bundle link is no longer in the up state. This behavior complies with the Class A bandwidth requirement defined in FRF.16.

The bundle interface's line protocol status is considered up when the Frame Relay data-link layer at the local router and peer device synchronize using the Local Management Interface (LMI), when LMI is enabled. The bundle line protocol remains up as long as the LMI keepalives are successful.

dMLFR Configuration Guidelines

To support dMLFR on the Cisco 7600 SIP-200, consider the following guidelines:

- dMLFR must be configured on the peer device.
- The dMLFR peer device must not send frames that require assembly.
- The Cisco 7600 SIP-200 supports distributed links under the following conditions:
  - All links are on the same Cisco 7600 SIP-200.
  - T1 and E1 links cannot be mixed in a bundle.
  - Member links in a bundle are recommended to have the same bandwidth.
- QoS is implemented on the Cisco 7600 SIP-200 for dMLFR.
- dMLFR is supported with Frame Relay over MPLS (FRoMPLS) on the Cisco 7600 SIP-200 between the customer edge (CE) and provider edge (PE) of the MPLS network.
- The Cisco 7600 SIP-200 only supports the RPR+ High Availability (HA) feature with dMLFR.
- dMLFR is supported in software by the Cisco 7600 SIP-200, or in hardware by the supported SPA. This support is determined by your link configuration.
- dMLFR is supported in software if bundle link members are on different SPAs in the same SIP.

Software-Based Guidelines

dMLFR will be implemented in the software if *any* of the following conditions are met:

- Any one bundle link member is a fractional T1 or E1 link.
- There are more than 12 T1 or E1 links in a bundle.

Hardware-Based Guidelines

dMLFR will be implemented in the hardware when *all* of the following conditions are met:

- All bundle link members are T1 or E1 only.
- All bundle links are on the same SPA.
- There are no more than 12 links in a bundle.

dMLFR Restrictions

When configuring dMLFR on the Cisco 7600 SIP-200, consider the following restrictions:

- FRF.9 hardware compression is not supported.
- Software compression is not supported.
- Encryption is not supported.
- The maximum differential delay supported is 50 ms when supported in hardware, and 100 ms when supported in software.
- Fragmentation is not supported on the transmit side.

dMLFR Configuration Tasks

The following sections describe how to configure dMLFR:

- Creating a Multilink Frame Relay Bundle, page 4-9 (required)
- Assigning an Interface to a dMLFR Bundle, page 4-10 (required)

Creating a Multilink Frame Relay Bundle

To configure the bundle interface for dMLFR, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface mfr** *number*	Configures a multilink Frame Relay bundle interface and enters interface configuration mode, where: • *number*—Specifies the number for the Frame Relay bundle.
Step 2	Router(config-if)# **frame-relay multilink bid** *name*	(Optional) Assigns a bundle identification name to a multilink Frame Relay bundle, where: • *name*—Specifies the name for the Frame Relay bundle. **Note** The bundle identification (BID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shutdown** and **no shutdown** commands in interface configuration mode.
Step 3	Router(config-if)# **frame-relay intf-type dce**	Configures the router to function as a digital communications equipment (DCE) device, or as a switch.

Assigning an Interface to a dMLFR Bundle

To configure an interface link and associate it as a member of a dMLFR bundle, use the following commands beginning in global configuration mode. Repeat these steps to assign multiple links to the dMLFR bundle.

Note If you use this task to assign more than 12 T1 or E1 interface links as part of the same bundle, or if any of the T1/E1 interface links are fractional T1/E1, or any links reside on multiple SPAs as part of the same bundle, then software-based dMLFR is implemented automatically by the Cisco 7600 SIP-200.

	Command	Purpose
Step 1	**1-Port Channelized OC-3/STM-1 SPA** Router(config)# **interface serial** *address* **2-Port and 4-Port Channelized T3 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***/***t1-number***:***channel-group* **8-Port Channelized T1/E1 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***:***channel-group*	Specifies a serial interface and enters interface configuration mode, where: • *address*—For the different supported syntax options for the *address* argument for the 1-Port Channelized OC-3/STM-1 SPA, refer to the "Interface Naming" section of the "Configuring the 1-Port Channelized OC-3/STM-1 SPA" chapter. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • *t1-number*—Specifies the logical T1 number in channelized mode. • *channel-group*—Specifies the logical channel group assigned to the time slots within the T1 or E1 group. **Note** If you configure a fractional T1/E1 interface on the SPA using a channel group and specify that fractional T1/E1 channel group as part of this task, then software-based dMLFR is implemented automatically by the Cisco 7600 SIP-200 when you assign the interface to the dMLFR bundle.
Step 2	Router(config-if)# **encapsulation frame-relay mfr** *number* [*name*]	Creates a multilink Frame Relay bundle link and associates the link with a bundle, where: • *number*—Specifies the number for the Frame Relay bundle. This number should match the dMLFR interface number specified in the **interface mfr** command. • *name*—(Optional) Specifies the name for the Frame Relay bundle.

	Command	Purpose
Step 3	Router(config-if)# **frame-relay multilink lid** *name*	(Optional) Assigns a bundle link identification name with a multilink Frame Relay bundle link, where: • *name*—Specifies the name for the Frame Relay bundle. **Note** The bundle link identification (LID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shutdown** and **no shutdown** commands in interface configuration mode.
Step 4	Router(config-if)# **frame-relay multilink hello** *seconds*	(Optional) Configures the interval at which a bundle link will send out hello messages, where: • *seconds*—Specifies the number of seconds between hello messages sent out over the multilink bundle. The default is 10 seconds.
Step 5	Router(config-if)# **frame-relay multilink ack** *seconds*	(Optional) Configures the number of seconds that a bundle link will wait for a hello message acknowledgment before resending the hello message, where: • *seconds*—Specifies the number of seconds a bundle link will wait for a hello message acknowledgment before resending the hello message. The default is 4 seconds.
Step 6	Router(config-if)# **frame-relay multilink retry** *number*	(Optional) Configures the maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment, where: • *number*—Specifies the maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment. The default is 2 tries.

Verifying dMLFR

To verify dMLFR configuration, use the **show frame-relay multilink** command. If you use the **show frame-relay multilink** command without any options, information for all bundles and bundle links is displayed.

The following examples show output for the **show frame-relay multilink** command with the **serial** *number* and **detailed** options. Detailed information about the specified bundle links is displayed.

```
Router# show frame-relay multilink serial6 detailed

Bundle: MFR49, State = down, class = A, fragmentation disabled
 BID = MFR49
 No. of bundle links = 1, Peer's bundle-id =
 Bundle links:

  Serial6/0/0:0, HW state = up, link state = Add_sent, LID = test
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = , RTT = 0 ms
    Statistics:
```

```
Add_link sent = 21, Add_link rcv'd = 0,
Add_link ack sent = 0, Add_link ack rcv'd = 0,
Add_link rej sent = 0, Add_link rej rcv'd = 0,
Remove_link sent = 0, Remove_link rcv'd = 0,
Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
Hello sent = 0, Hello rcv'd = 0,
Hello_ack sent = 0, Hello_ack rcv'd = 0,
outgoing pak dropped = 0, incoming pak dropped = 0
```

Configuring Distributed Multilink PPP on the Cisco 7600 SIP-200

The Distributed Multilink Point-to-Point Protocol (dMLPPP) feature allows you to combine T1/E1 lines into a bundle that has the combined bandwidth of multiple T1/E1 lines. This is done by using a dMLPPP link. You choose the number of bundles and the number of T1/E1 lines in each bundle. This allows you to increase the bandwidth of your network links beyond that of a single T1/E1 line without having to purchase a T3 line.

Note Based on your link configuration, dMLPPP can be either software-based on the Cisco 7600 SIP-200, or hardware-based on the 8-Port Channelized T1/E1 SPA and 2-Port and 4-Port Channelized T3 SPAs. For more information about the hardware-based configuration, see also Chapter 17, "Configuring the 8-Port Channelized T1/E1 SPA," Chapter 19, "Configuring the 2-Port and 4-Port Channelized T3 SPAs.", and Chapter 25, "configuring the 1-Port Channelized OC3/STM-1 SPA.

Table 4-3 provides information about where the dMLppp feature for SPA interfaces is supported.

Table 4-3 dMLPPP Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Hardware-based dMLPPP	Supported	Not supported.	Not supported.
Hardware- and software-based dMLPPP	In Cisco IOS Release 12.2(18)SXE and later: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA In Cisco IOS Release 12.2(33)SRA and later: • 1-Port Channelized OC3/STM-1 SPA	Not supported.	Not supported.

This section includes the following topics:

- dMLPPP Configuration Guidelines, page 4-12
- dMLPPP Configuration Tasks, page 4-13
- Verifying dMLPPP, page 4-17

dMLPPP Configuration Guidelines

dMLPPP is supported in software by the Cisco 7600 SIP-200, or in hardware by the supported SPA. This support is determined by your link configuration.

The Cisco 7600 SIP-200 supports distributed links under the following conditions:

- All links are on the same Cisco 7600 SIP-200.

- T1 and E1 links cannot be mixed in a bundle.
- Member links in a bundle are recommended to have the same bandwidth.
- QoS is implemented on the Cisco 7600 SIP-200 for dMLPPP.

Software-Based Guidelines

dMLPPP will be implemented in the software if *any* of the following conditions are met:

- Any one bundle link member is a fractional T1 or E1 link.
- There are more than 12 T1 or E1 links in a bundle.
- To enable fragmentation for software-based dMLPPP, you must configure the **ppp multilink interleave** command. This command is not required to enable fragmentation for hardware-based dMLPPP.

Hardware-Based Guidelines

dMLPPP will be implemented in the hardware when *all* of the following conditions are met:

- All bundle link members are T1 or E1 only.
- All bundle links are on the same SPA.
- There are no more than 12 links in a bundle.

dMLPPP Restrictions

When configuring dMLPPP on the Cisco 7600 SIP-200, consider the following restrictions:

- Hardware and software compression is not supported.
- Encryption is not supported.
- The maximum differential delay supported is 50 ms when supported in hardware, and 100 ms when supported in software.

dMLPPP Configuration Tasks

The following sections describe how to configure dMLPPP:

- Enabling Distributed CEF Switching, page 4-13 (required)
- Creating a dMLPPP Bundle, page 4-14 (required)
- Assigning an Interface to a dMLPPP Bundle, page 4-16 (required)
- Configuring Link Fragmentation and Interleaving over dMLPPP, page 4-17 (optional)

Enabling Distributed CEF Switching

To enable dMLPPP, you must first enable distributed CEF switching. Distributed CEF switching is enabled by default on the Cisco 7600 series router.

To enable dCEF, use the following command in global configuration mode:

Command	Purpose
Router(config)# **ip cef distributed**	Enables distributed CEF switching.

Creating a dMLPPP Bundle

To configure a dMLPPP bundle, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface multilink** *group-number*	Creates a multilink interface and enters interface configuration mode, where: • *group-number*—Specifies the group number for the multilink bundle.
Step 2	Router(config-if)# **ip address** *ip-address mask*	Sets the IP address for the multilink group, where: • *ip-address*—Specifies the IP address for the interface. • *mask*—Specifies the mask for the associated IP subnet.
Step 3	Router(config-if)# **ppp multilink interleave**	(Optional—Software-based LFI) Enables fragmentation for the interfaces assigned to the multilink bundle. Fragmentation is disabled by default in software-based LFI.
Step 4	Router(config-if)# **ppp multilink mrru** [**local** | **remote**] *mrru-value*	Configures the MRRU value negotiated on a multilink bundle when MLP is used. • **local**—(Optional) Configures the local MRRU value. The default values for the local MRRU are the value of the multilink group interface MTU for multilink group members, and 1524 bytes for all other interfaces. • **remote**—(Optional) Configures the minimum value that software will accept from the peer when it advertises its MRRU. By default, the software accepts any peer MRRU value of 128 or higher. You can specify a higher minimum acceptable MRRU value in a range from 128 to 16384 bytes.

	Command	Purpose
Step 5	Router(config-if)# **mtu** *bytes*	(Optional) Adjusts the maximum packet size or MTU size. • Once you configure the MRRU on the bundle interface, you enable the router to receive large reconstructed MLP frames. You may want to configure the bundle MTU so the router can transmit large MLP frames, although it is not strictly necessary. • The maximum recommended value for the bundle MTU is the value of the peer's MRRU. The default MTU for serial interfaces is 1500. The software will automatically reduce the bundle interface MTU if necessary, to avoid violating the peer's MRRU.
Step 6	Router(config-if)# **ppp multilink fragment delay** *delay*	(Optional) Sets the fragmentation size satisfying the configured delay on the multilink bundle, where: • *delay*—Specifies the delay in milliseconds.

Assigning an Interface to a dMLPPP Bundle

To configure an interface PPP link and associate it as a member of a multilink bundle, use the following commands beginning in global configuration mode. Repeat these steps to assign multiple links to the dMLPPP bundle.

Note If you use this task to assign more than 12 T1 or E1 interface links as part of the same bundle, or if any of the T1/E1 interface links are fractional T1/E1, or any links reside on multiple SPAs as part of the same bundle, then software-based dMLPPP is implemented automatically by the Cisco 7600 SIP-200.

	Command	Purpose
Step 1	**1-Port Channelized OC-3/STM-1 SPA** Router(config)# **interface serial** *address* **2-Port and 4-Port Channelized T3 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***/***t1-number***:***channel-group* **8-Port Channelized T1/E1 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***:***channel-group*	Specifies a serial interface and enters interface configuration mode, where: • *address*—For the different supported syntax options for the *address* argument for the 1-Port Channelized OC-3/STM-1 SPA, refer to the “Interface Naming” section of the “Configuring the 1-Port Channelized OC-3/STM-1 SPA” chapter. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • *t1-number*—Specifies the logical T1 number in channelized mode. • *channel-group*—Specifies the logical channel group assigned to the time slots within the T1 or E1 group. **Note** If you configure a fractional T1/E1 interface on the SPA using a channel group and specify that fractional T1/E1 channel group as part of this task, then software-based dMLPPP is implemented automatically by the Cisco 7600 SIP-200 when you assign the interface to the dMLPPP bundle.
Step 2	Router(config-if)# **encapsulation ppp**	Enables PPP encapsulation.
Step 3	Router(config-if)# **ppp multilink**	(Optional) Enables dMLPPP on the interface.
Step 4	Router(config-if)# **ppp authentication chap**	(Optional) Enables Challenge Handshake Authentication Protocol (CHAP) authentication.

	Command	Purpose
Step 5	Router(config-if)# **ppp chap hostname** *name*	(Optional) Assigns a name to be sent in the CHAP challenge. • *name*—Specifies an alternate username that will be used for CHAP authentication
Step 6	Router(config-if)# **ppp multilink group** *group-number*	Assigns the interface to a multilink bundle, where: • *group-numbe*r—Specifies the group number for the multilink bundle. This number should match the dMLPPP interface number specified in the **interface multilink** command.

Configuring Link Fragmentation and Interleaving over dMLPPP

Link fragmentation and interleaving (LFI) over dMLPPP is supported in software on the Cisco 7600 SIP-200, or in hardware on the 2-Port and 4-Port Channelized T3 SPA and the 8-Port Channelized T1/E1 SPA. This support is determined by your link configuration.

Software-Based Guidelines

When configuring LFI over dMLPPP, consider the following guidelines for software-based LFI:

- LFI over dMLPPP will be configured in software if there is more than one link assigned to the dMLPPP bundle.
- LFI is disabled by default in software-based LFI. To enable LFI on the multilink interface, use the **ppp multilink interleave** command.
- Fragmentation size is calculated from the delay configured and the member link bandwidth.
- You must configure a policy map with a class under the multilink interface.
- CRTP should not be configured on a multilink interface when LFI is enabled on the multilink interface if the multilink bundle has more than one member link, and a QoS policy with a feature is enabled on the multilink interface.

Hardware-Based Guidelines

When configuring LFI over dMLPPP, consider the following guidelines for hardware-based LFI:

- LFI over dMLPPP will be configured in hardware if you only assign one link (either T1/E1 or fractional T1/E1) to the dMLPPP bundle.
- LFI is enabled by default in hardware-based LFI with a default size of 512 bytes. To enable LFI on the serial interface, use the **ppp multilink interleave** command.
- A policy map having a class needs to be applied to the multilink interface.

Verifying dMLPPP

To verify dMLPPP configuration, use the **show ppp multilink** command, as shown in the following example:

```
Router# show ppp multilink

Multilink2, bundle name is group2
  Bundle up for 00:01:21
  Bundle is Distributed
  0 lost fragments, 0 reordered, 0 unassigned
  0 discarded, 0 lost received, 1/255 load
```

```
  0x0 received sequence, 0x0 sent sequence
  Member links: 2 active, 0 inactive (max not set, min not set)
    Se4/3/0/1:0, since 00:01:21, no frags rcvd
    Se4/3/0/1:1, since 00:01:19, no frags rcvd
```

If hardware-based dMLPPP is configured on the SPA, the **show ppp multilink** command displays "Multilink in Hardware" as shown in the following example:

```
Router# show ppp multilink

Multilink1, bundle name is group1
  Bundle up for 00:00:13
  Bundle is Distributed
  0 lost fragments, 0 reordered, 0 unassigned
  0 discarded, 0 lost received, 206/255 load
  0x0 received sequence, 0x0 sent sequence
  Member links: 2 active, 0 inactive (max not set, min not set)
    Se4/2/0/1:0, since 00:00:13, no frags rcvd
    Se4/2/0/2:0, since 00:00:10, no frags rcvd
  Distributed fragmentation on. Fragment size 512.  Multilink in Hardware.
```

Configuring Distributed Link Fragmentation and Interleaving for Frame Relay and ATM Interfaces

The Distributed Link Fragmentation and Interleaving (dLFI) feature supports the transport of real-time traffic, such as voice, and non-real-time traffic, such as data, on lower-speed Frame Relay and ATM virtual circuits (VCs) and on leased lines without causing excessive delay to the real-time traffic.

This feature is implemented using dMLPPP over Frame Relay, ATM, and leased lines. The feature enables delay-sensitive real-time packets and non-real-time packets to share the same link by fragmenting the large data packets into a sequence of smaller data packets (fragments). The fragments are then interleaved with the real-time packets. On the receiving side of the link, the fragments are reassembled and the packets reconstructed.

The dLFI feature is often useful in networks that send real-time traffic using Distributed Low Latency Queueing, such as voice, but have bandwidth problems that delay this real-time traffic due to the transport of large, less time-sensitive data packets. The dLFI feature can be used in these networks to disassemble the large data packets into multiple segments. The real-time traffic packets then can be sent between these segments of the data packets. In this scenario, the real-time traffic does not experience a lengthy delay waiting for the low- data packets to traverse the network. The data packets are reassembled at the receiving side of the link, so the data is delivered intact.

The ability to configure Quality of Service (QoS) using the Modular QoS CLI while also using dMLPPP is also introduced as part of the dLFI feature.

For specific information about configuring dLFI, refer to the *FlexWAN and Enhanced FlexWAN Module Installation and Configuration Note* located at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/cfgnotes/flexport/combo/index.htm

For information about configuring dLFI on ATM SPAs, see the "Configuring Link Fragmentation and Interleaving with Virtual Templates" section on page 7-46 in Chapter 7, "Configuring the ATM SPAs."

Table 4-4 provides information about where the dLFI feature for SPA interfaces is supported.

Table 4-4 dLFI Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Hardware-based dLFI	In Cisco IOS Release 12.2(18)SXE and later: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA	In Cisco IOS Release 12.2(18)SXE and later: • 2-Port OC-3c/STM-1 ATM SPA • 1-Port OC-12c/STM-4 ATM SPA	Not supported.
Hardware- and software-based dLFI	In Cisco IOS Release 12.2(33)SRA: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA • 1-Port Channelized OC-3/STM-1 SPA	Not supported.	Not supported.
dLFI with MPLS	Not supported.	Not supported.	Not supported.
dLFI with MPLS on VPN	Supported between the CE and PE devices, and with virtual routing and forwarding (VRF) configuration.	Not supported.	Not supported.

Cisco 7600 Series Router LFI Restrictions

When configuring LFI on the Cisco 7600 series router, consider the following restrictions:

- A maximum number of 200 permanent virtual circuits (PVCs) or switched virtual circuits (SVCs) using Link Fragmentation and Interleaving (LFI) is supported for all ATM SPAs (or other ATM modules) in a Cisco 7600 series router.
- LFI using FRF.12 is supported in hardware only for the 2-Port and 4-Port Channelized T3 SPA and 8-Port Channelized T1/E1 SPA.
- LFI over dMLPPP is supported in software or hardware depending on your link configuration. For more information about software-based LFI over dMLPPP, see the "Configuring Link Fragmentation and Interleaving over dMLPPP" section on page 4-17. For more information about hardware-based LFI over dMLPPP, refer to the Chapter 17, "Configuring the 8-Port Channelized T1/E1 SPA," and Chapter 19, "Configuring the 2-Port and 4-Port Channelized T3 SPAs."
- QoS is implemented on the Cisco 7600 SIP-200 for dLFI.

Configuring Voice over Frame Relay FRF.11 and FRF.12

Voice over Frame Relay (VoFR) enables a router to carry voice traffic (for example, telephone calls and faxes) over a Frame Relay network using the FRF.11 protocol. This specification defines multiplexed data, voice, fax, dual-tone multi-frequency (DTMF) digit-relay, and channel-associated signaling (CAS)/robbed-bit signaling frame formats. The Frame Relay backbone must be configured to include the map class and Local Management Interface (LMI).

The Cisco VoFR implementation enables dynamic- and tandem-switched calls and Cisco trunk calls. Dynamic-switched calls have dial-plan information included that processes and routes calls based on the telephone numbers. The dial-plan information is contained within dial-peer entries.

Note Because the Cisco 7600 series router does not support voice modules, the Cisco 7600 series router can act only as a VoFR tandem switch when FRF.11 or FRF.12 is configured on the SIPs.

Tandem-switched calls are switched from incoming VoFR to an outgoing VoFR-enabled data-link connection identifier (DLCI) and tandem nodes enable the process. The nodes also switch Cisco trunk calls.

Permanent calls are processed over Cisco private-line trunks and static FRF.11 trunks that specify the frame format and coder types for voice traffic over a Frame Relay network.

VoFR connections depend on the hardware platform and type of call. The types of calls are:

- Switched (user dialed or auto-ringdown and tandem)
- Permanent (Cisco trunk or static FRF.11 trunk)

Note FRF.11 support was removed in Cisco IOS Release 12.2(18)SXF on the Cisco 7600 series router.

Table 4-5 provides information about where the VoFR feature for SPA interfaces is supported.

Table 4-5 VoFR Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
FRF.11	In Cisco IOS Releases 12.2(18)SXE and 12.2(18)SXE2: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA	Not supported.	Not supported.
FRF.12	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA: • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port Channelized T3 SPA • 1-Port Channelized OC-3/STM-1 SPA	Not supported.	Not supported.

For specific information about configuring voice over Frame Relay FRF.11 and FRF.12, refer to the *Cisco IOS Voice, Video, and Fax Configuration Guide* located at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fvvfax_c/vvfvofr.htm

Configuring Layer 2 Interworking Features on a SIP

This section provides SIP-specific information about configuring the Layer 2 interworking features on the Cisco 7600 series router. It includes the following topics:

- Configuring Bridging for ATM Interfaces (RFC 1483/RFC 2684), page 4-21
- Configuring Multipoint Bridging, page 4-23
- Configuring PPP Bridging Control Protocol Support, page 4-35

Configuring Bridging for ATM Interfaces (RFC 1483/RFC 2684)

The following types of bridging are supported on ATM SPAs in the Cisco 7600 series router. For information about SIP and SPA compatibility with each of these features, see Table 4-6.

Note RFC 1483 has been obsoleted and superseded by RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*. To avoid confusion, this document continues to refer to the original RFC numbers.

- RFC 1483/RFC 2684 bridging for point-to-point PVCs —RFC 1483 has been obsoleted and superseded by RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*. RFC 2684 specifies the implementation of point-to-point bridging of Layer 2 PDUs from an ATM interface.
- RFC 1483/RFC 2684 bridging with IEEE 802.1Q tunneling—Allows service providers to aggregate multiple VLANs over a single VLAN, while still keeping the individual VLANs segregated and preserving the VLAN IDs for each customer. This tunneling simplifies traffic management for the service provider, while keeping customer networks secure.
- RFC 1483/RFC 2684 half-bridging—Routes IP traffic from a stub-bridged Ethernet LAN over a bridged RFC 1483/RFC 2684 ATM interface, without using integrated routing and bridging (IRB). This allows bridged traffic that terminates on an ATM PVC to be routed on the basis of the destination IP address.
- ATM routed bridge encapsulation (RBE)—The ATM SPAs support ATM Routed Bridge Encapsulation (RBE), which is similar in functionality to RFC 1483 ATM half-bridging, except that ATM half-bridging is configured on a point-to-multipoint PVC, while RBE is configured on a point-to-point PVC.
- Bridging of routed encapsulations (BRE)—Enables an ATM SPA to receive RFC 1483/2684 routed encapsulated packets and forward them as Layer 2 frames. In a BRE configuration, the PVC receives the routed PDUs, removes the RFC 1483 routed encapsulation header, and adds an Ethernet MAC header to the packet. The Layer 2 encapsulated packet is then switched by the forwarding engine to the Layer 2 interface determined by the VLAN number and destination MAC.
- Per VLAN Spanning Tree (PVST) to PVST+ Bridge Protocol Data Unit (BPDU) interoperability—PVST is a Cisco proprietary protocol that allows a Cisco device to support multiple spanning tree topologies on a per-VLAN basis. PVST uses the BPDUs defined in IEEE 802.1D, but instead of one STP instance per switch, there is one STP instance per VLAN. PVST+ is a Cisco proprietary protocol that creates one STP instance per VLAN (as in PVST). However, PVST+ enhances PVST and uses Cisco proprietary BPDUs with a special 802.2 Subnetwork Access Protocol (SNAP) Organizational Unique Identifier (OUI) instead of the standard IEEE 802.1D frame format used by PVST. PVST+ BPDUs are also known as Simple Symmetric Transmission Protocol (SSTP) BPDUs.

Table 4-6 provides information about where the bridging features for ATM SPA interfaces are supported. For more details about the implementation and information about configuring bridging for ATM SPA interfaces, see Chapter 7, "Configuring the ATM SPAs."

Table 4-6 Bridging for ATM Interfaces Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
RFC 1483/RFC 2684 Bridging for Point-to-Point PVCs (**bridge-domain** command)	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA • 1-Port OC-12c/STM-4 ATM SPA	Not supported.
RFC 1483/RFC 2684 Bridging with IEEE 802.1Q Tunneling for Point-to-Point PVCs (**bridge-domain dot1q-tunnel** command)	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA and later: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA • 1-Port OC-12c/STM-4 ATM SPA In Cisco IOS Release 12.2(18)SXF and Cisco IOS Release 12.2(33)SRA and later: • 1-Port OC-48c/STM-16 ATM SPA	Not supported.
RFC 1483/RFC 2684 Half-Bridging for Point-to-Multipoint PVCs	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA	Not supported.	Not supported.
RFC 1483/RFC 2684 Routed Bridge Encapsulation (RBE) for Point-to-Point PVCs	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA	Not supported.	Not supported.
RFC 1483/RFC 2684 Bridging of Routed Encapsulations (BRE) for PVCs	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA	Not supported.	Not supported.

Table 4-6 Bridging for ATM Interfaces Feature Compatibility by SIP and SPA Combination (continued)

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Enhancements to RFC 1483/RFC 2684 Spanning Tree Interoperability (PVST to PVST+ BPDU Interoperability)	In Cisco IOS Release 12.2(18)SXF2 and later, and in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA	In Cisco IOS Release 12.2(18)SXF2 and later, and in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA • 1-Port OC-12c/STM-4 ATM SPA • 1-Port OC-48c/STM-16 ATM SPA	Not supported.
Multi-VLAN to VC	In Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA and later: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA	In Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA • 1-Port OC-12c/STM-4 ATM SPA • 1-Port OC-48c/STM-16 ATM SPA	Not supported.

Configuring Multipoint Bridging

Multipoint bridging (MPB) enables the connection of multiple ATM PVCs, Frame Relay PVCs, Bridge Control Protocol (BCP) ports, and WAN Gigabit Ethernet subinterfaces into a single broadcast domain (virtual LAN), together with the LAN ports on that VLAN. This enables service providers to add support for Ethernet-based Layer 2 services to the proven technology of their existing ATM and Frame Relay legacy networks. Customers can then use their current VLAN-based networks over the ATM or Frame Relay cloud. This also allows service providers to gradually update their core networks to the latest Gigabit Ethernet optical technologies, while still supporting their existing customer base.

ATM interfaces use RFC 1483/RFC 2684 bridging, and Frame Relay interfaces use RFC 1490/RFC 2427 bridging, both of which provide an encapsulation method to allow the transport of Ethernet frames over each type of Layer 2 network.

Beginning in Cisco IOS Release 12.2(33)SRA, MPB support is added on the Cisco 7600 SIP-400 to multiplex different VLANs that are configured across multiple Gigabit Ethernet subinterfaces into a single broadcast domain. Gigabit Ethernet interfaces can also reside on different Cisco 7600 SIP-400s and belong to the same bridge domain.

Table 4-7 provides information about where the MPB features for SPA interfaces are supported.

Table 4-7 MPB Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
MPB—60 VCs or interfaces per VLAN globally in system	In Cisco IOS Release 12.2(18)SXE and later: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA • 2-Port and 4-Port Channelized T3 SPA • 2-Port and 4-Port Clear Channel T3/E3 SPA • 8-Port Channelized T1/E1 SPA	In Cisco IOS Release 12.2(18)SXE and later: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA • 1-Port OC-12c/STM-4 ATM SPA	Not supported.
MPB—112 VCs or interfaces per VLAN on each SIP **Note** If you are using Virtual Private LAN Service (VPLS), see the MPB configuration guidelines.	In Cisco IOS Release 12.2(33)SRA: • 1-Port Channelized OC-3/STM-1 SPA • 2-Port and 4-Port OC-3c/STM-1 ATM SPA • 2-Port and 4-Port OC-3c/STM-1 POS SPA • 2-Port and 4-Port Channelized T3 SPA • 2-Port and 4-Port Clear Channel T3/E3 SPA • 8-Port Channelized T1/E1 SPA	Not applicable.	Not supported.
MPB—120 VCs or interfaces per VLAN on each SIP **Note** If you are using VPLS, see the MPB bridging configuration guidelines.	Not supported.	In Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-3c/STM-1 ATM SPA • 1-Port OC-12c/STM-4 ATM SPA • 1-Port OC-48c/STM-16 ATM SPA • 2-Port and 4-Port OC-3c/STM-1 POS SPA • 1-Port OC-12c/STM-4 POS SPA • 1-Port OC-48c/STM-16 POS SPA	Not supported.

Table 4-7 MPB Feature Compatibility by SIP and SPA Combination (continued)

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
MPB on Gigabit Ethernet—Layer 2 bridging of frames between subinterfaces on different physical Gigabit Ethernet ports	Not supported.	In Cisco IOS Release 12.2(33)SRA: • 2-Port Gigabit Ethernet SPA	Not supported.
PIM snooping for MPB	Not supported.	Supported for all SPAs in Cisco IOS Release 12.2(33)SRA.	Not supported.

Configuring MPB for ATM PVCs

You can configure MPB manually on individual PVCs, or you can configure a range of PVCs to configure all of the PVCs at one time. ATM interfaces use RFC 1483/RFC 2684 bridging, which provides an encapsulation method to allow the transport of Ethernet frames over the Layer 2 network.

Note RFC 1483 has been obsoleted and superseded by RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*. To avoid confusion, this document continues to refer to the original RFC numbers.

MPB for ATM PVCs Configuration Guidelines

- Only ATM permanent virtual circuits (PVCs) are supported. SVCs are not supported.
- MPB is not supported on VLAN IDs 0, 1, 1002–1005, and 4095.
- Refer to Table 4-7 for limitations on the number of supported VCs.
- If you are using VPLS on a VC, then the total number of supported VC connection points for MPB (112 for the Cisco 7600 SIP-200, or 120 for the Cisco 7600 SIP-400) is reduced by one for each VPLS VC configured on *that* bridged VLAN. This reduces the total available number of VC connection points for MPB on that VLAN globally for that SIP. For example, if you configure 10 VPLS VCs on bridged VLAN 100, for a SPA on a Cisco 7600 SIP-200 in slot 4, then 10 connection points are allocated to the VPLS VCs for VLAN 100 across the SIP in slot 4. The total number of connection points available for MPB on VLAN 100 for the Cisco 7600 SIP-200 in slot 4 is 112 minus 10, or 102. A different VLAN (for example, VLAN 300) on that same Cisco 7600 SIP-200 in slot 4, without any VPLS VCs, will have the full 112 VCs available.
- Routing and bridging is supported on the same interface or subinterface, but for security reasons, routing and bridging is not supported on any given PVC. Therefore, you should not configure an IP address on a point-to-point subinterface and then configure bridging on a PVC on that subinterface.
- For a limited form of trunking on ATM PVCs supporting multiple VLANs to a single VC, you can configure **dot1q** *tag*. However, this configuration can lead to a performance penalty. When using this configuration, you can specify up to 32 **bridge-domain** command entries for a single PVC. The highest tag value in a group of **bridge-domain** commands must be greater than the first tag entered (but less than 32 greater than the first tag entered).

To configure MPB for ATM PVCs, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **vlan** {*vlan-id* | *vlan-range*}	Adds the specified VLAN IDs to the VLAN database and enters VLAN configuration mode, where: • *vlan-id*—Specifies a single VLAN ID. The valid range is from 2 to 4094. • *vlan-range*—Specifies multiple VLAN IDs, as either a list or a range. The *vlan-range* can contain a list of the VLAN IDs, separated by a comma (,), dash (-), or both. **Note** Before you can use a VLAN for multipoint bridging, you must manually enter its VLAN ID into the VLAN database.
Step 2	Router(config)# **interface atm** *slot***/***subslot***/***port*	Specifies or creates an ATM interface, where: • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 3	Router(config)# **interface atm** *slot***/***subslot***/***port***.***subinterface* {**point-to-point** | **multipoint**}	Specifies or creates a subinterface and enters subinterface configuration mode, where: • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • **.***subinterface*—Specifies the number of the subinterface on the interface port. • **point-to-point**—Specifies a point-to-point subinterface. • **multipoint**—Specifies a multipoint subinterface that allows multiple PVCs to use the same subinterface.
Step 4	Router(config-subif)# **no ip address**	Disables IP processing on the subinterface by removing its IP address.

	Command	Purpose
Use the following commands (**pvc** and **bridge-domain**) to create and configure PVCs individually. Repeat these commands as desired. Or, use the **range pvc** and **bridge-domain** command with the **increment** keyword to configure a range of PVCs.		
Step 5	Router(config-subif)# **pvc** [*name*] *vpi*/*vci* or Router(config-subif)# **range** [*range-name*] **pvc** *start-vpi*/*start-vci end-vpi*/*end-vci*	Configures a new ATM PVC or range of ATM PVCs with the specified VPI and VCI numbers and enters VC configuration mode or PVC range configuration mode, where: • *name*—(Optional) Specifies the descriptive name to identify this PVC. • *vpi/vci*—Specifies the virtual path identifier (VPI) and virtual channel identifier (VCI) for this PVC. • *range-name*—(Optional) Specifies the descriptive name of the range, up to a maximum of 15 characters. • *start-vpi*/—Specifies the beginning value for the range of virtual path identifiers (VPIs). The valid range is from 0 to 255, with a default of 0. • *start-vci*—Specifies the beginning value for a range of virtual channel identifiers (VCIs). The valid range is from 32 to 65535. • *end-vpi*/—Specifies the end value for the range of VPIs. The valid range is from 0 to 255, with a default that is equal to the *start-vpi* value. • *end-vci*—Specifies the end value for a range of virtual channel identifiers (VCIs). The VCI value ranges from 32 to 65535.

	Command	Purpose
Step 6	Router(config-if-atm-vc)# **bridge-domain** *vlan-id* [**access** | **dot1q** [*tag*]| **dot1q-tunnel**] [**ignore-bpdu-pid**] [**pvst-tlv** *CE-vlan*] [**increment**] [**split-horizon**]	Enables RFC 1483 bridging to map a bridged VLAN to an ATM PVC, where: • *vlan-id*—Specifies the number of the VLAN to be used in this bridging configuration. The valid range is from 2 to 4094. The VLAN ID must have been previously added to the VLAN database in Step 1. • **access**—(Optional) Enables access-only bridging access mode, in which the bridged connection does not transmit or act upon bridge protocol data unit (BPDU) packets. • **dot1q**—(Optional) Enables IEEE 802.1Q tagging to preserve the class of service (CoS) information from the Ethernet frames across the ATM network. If not specified, the ingress side assumes a CoS value of 0 for QoS purposes. Using the **dot1q** keyword helps avoid misconfiguration because incoming untagged frames, or tagged frames that don't match the specified *vlan-id* are dropped. • *tag*—(Optional—ATM PVCs only) Specifies the IEEE 802.1Q value in the range 1 to 4095. You can specify up to 32 **bridge-domain** command entries using **dot1q** *tag* for a single PVC. The highest tag value in a group of **bridge-domain** commands must be greater than the first tag entered (but less than 32 greater than the first tag entered). • **dot1q-tunnel**—(Optional) Enables IEEE 802.1Q tunneling mode, so that service providers can use a single VLAN to support customers who have multiple VLANs, while preserving customer VLAN IDs and keeping traffic in different customer VLANs segregated. **Note** The **access**, **dot1q**, and **dot1q-tunnel** options are mutually exclusive. If you do not specify any of these options, the connection operates in "raw" bridging access mode, which is similar to **access**, except that the connection processes and transmits BPDU packets. • **ignore-bpdu-pid**—(Optional—ATM PVCs only) Ignores the protocol-ID field in RFC 1497 bridge protocol data unit (BPDU) packets, to allow interoperation with ATM customer premises equipment (CPE) devices that do not distinguish BPDU packets from data packets.

Command	Purpose
	• **pvst-tlv** *CE-vlan*—(Optional) When transmitting, translates PVST+ BPDUs into IEEE BPDUs. When receiving, translates IEEE BPDUs into PVST+ BPDUs. *CE-vlan* specifies the customer-edge VLAN in the SSTP Tag-Length-Value (TLV) to be inserted in an IEEE BPDU to a PVST+ BPDU conversion. • **increment**—(Optional—PVC range configuration mode only) Increments the bridge domain number for each PVC in the range. This keyword is used when you are configuring a range of PVCs using the **range pvc** command. • **split-horizon**—(Optional) Drops egress traffic going out a VC or interface with split-horizon configured, that arrived on an interface with split-horizon configured.

Verifying MPB for ATM PVCs

To display information about the PVCs that have been configured on ATM interfaces, use the following commands:

- **show atm pvc**—Displays a summary of the PVCs that have been configured.
- **show atm vlan**—Displays the connections between PVCs and VLANs.

> **Note** Use the **show atm vlan** command instead of the **show interface trunk** command to display information about ATM interfaces being used for multipoint bridging.

The following shows an example of each command:

```
Router# show atm pvc

            VCD /                                                  Peak  Avg/Min Burst
Interface   Name          VPI     VCI   Type     Encaps    SC   Kbps   Kbps    Cells   Sts
5/0/0       1               0      102  PVC      SNAP      UBR  599040                 UP
5/0/0       2               0      103  PVC      SNAP      UBR  599040                 UP
5/0/0       3               0      111  PVC      SNAP      UBR  599040                 UP
5/0/0       3               0      111  PVC      SNAP      UBR  599040                 UP
5/0/0       3               0      111  PVC      SNAP      UBR  599040                 UP

Router# show atm vlan

Options Legend: DQ - dot1q;  DT - dot1q-tunnel;  MD - multi-dot1q;
                AC - access; SP - split-horizon; BR - broadcast;
                IB - ignore-bpdu-pid;
                DEF - default

Interface             VCD    VPI        Network  Customer   PVC       Options
                             /VCI       Vlan ID  Dot1Q-ID   Status
ATM5/0/0               1     0/102        102     1002        UP      MD
ATM5/0/0               2     0/103        103     1003        UP      MD
```

```
ATM5/0/0              3     0/111        111    1111       UP     MD
ATM5/0/0              3     0/111        112    1112       UP     MD
ATM5/0/0              3     0/111        113    1113       UP     MD
```

Configuring MPB for Frame Relay

You can configure MPB for Frame Relay on individual DLCI circuits. You can optionally add 802.1Q tagging or 802.1Q tunneling. Frame Relay interfaces use RFC 1490/RFC 2427 bridging, which provides an encapsulation method to allow the transport of Ethernet frames over the Layer 2 network.

Note RFC 1490 has been obsoleted and superseded by RFC 2427, *Multiprotocol Interconnect over Frame Relay*. To avoid confusion, this document continues to refer to the original RFC numbers.

MPB for Frame Relay Configuration Guidelines

- Multipoint bridging on Frame Relay interfaces supports only IETF encapsulation. Cisco encapsulation is not supported for MPB.
- MPB is not supported on VLAN IDs 0, 1, 1002–1005, and 4095.
- Refer to Table 4-7 for limitations on the number of supported VCs.
- If you are using VPLS, then the total number of supported DLCI connection points for MPB (112 for the Cisco 7600 SIP-200, or 120 for the Cisco 7600 SIP-400) is reduced by one for each VPLS instance configured on *that* bridged VLAN. This reduces the total available number of DLCI connection points for MPB on that VLAN globally for that SIP. For example, if you configure 10 VPLS instances on a bridged VLAN 100, for a SPA on a Cisco 7600 SIP-200 in slot 4, then 10 connection points are allocated to the VPLS instances for VLAN 100 across the SIP in slot 4. The total number of connection points available for MPB on VLAN 100 for the Cisco 7600 SIP-200 in slot 4 is 112 minus 10, or 102. A different VLAN (for example, VLAN 300) on that same Cisco 7600 SIP-200 in slot 4, without any VPLS DLCIs, will have the full 112 DLCIs available.
- Routing and bridging is supported on the same interface or subinterface, but for security reasons, routing and bridging is not supported on any given DLCI. Therefore, you should not configure an IP address on a point-to-point subinterface and then configure bridging on a DLCI on that subinterface.

To configure MPB for Frame Relay on serial or POS SPAs, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **vlan** {*vlan-id* | *vlan-range*}	Adds the specified VLAN IDs to the VLAN database and enters VLAN configuration mode, where: • *vlan-id*—Specifies a single VLAN ID. The valid range is from 2 to 4094. • *vlan-range*—Specifies multiple VLAN IDs, as either a list or a range. The *vlan-range* can contain a list of the VLAN IDs, separated by a comma (,), dash (-), or both. **Note** Before you can use a VLAN for multipoint bridging, you must manually enter its VLAN ID into the VLAN database.
Step 2	Router(config)# **interface serial** *slot***/***subslot***/***port* or Router(config)# **interface pos** *slot***/***subslot***/***port*	Specifies or creates a serial or POS interface, where: • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 3	Router(config-if) **encapsulation frame-relay ietf**	Enables Frame Relay encapsulation on the interface, using IETF encapsulation. You must specify the **ietf** keyword either here or in Step 6 for each individual DLCI. **Note** Multipoint bridging does not support Cisco encapsulation using the **cisco** keyword.
Step 4	**2-Port and 4-Port Clear Channel T3/E3 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***.***subinterface* {**point-to-point** | **multipoint**} **2-Port and 4-Port Channelized T3 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***/***t1-number***:***channel-group***.***subinterface* {**point-to-point** | **multipoint**} **8-Port Channelized T1/E1 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***:***channel-group***.***subinterface* {**point-to-point** | **multipoint**}	Specifies or creates a subinterface and enters subinterface configuration mode, where: • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • **.***subinterface*—Specifies the number of the subinterface on the interface port. • *t1-number*—Specifies the logical T1 number in channelized mode.

	Command	Purpose
	1-Port OC-12c/STM-4 POS SPA or 2-Port and 4-Port OC-3c/STM-1 POS SPA Router(config)# **interface pos** *slot***/***subslot***/***port***.***subinterface* {**point-to-point** | **multipoint**}	• *channel-group*—Specifies the logical channel group assigned to the time slots within the T1 or E1 group. • **point-to-point**—Specifies a point-to-point subinterface. • **multipoint**—Allows multiple PVCs to use the same subinterface.
Step 5	Router(config-subif)# **no ip address**	Disables IP processing on a particular interface by removing its IP address.
Step 6	Router(config-subif)# **frame-relay interface-dlci** *dlci* [**ietf**]	Creates the specified DLCI on the subinterface and enters DLCI configuration mode, where: • *dlci*—Specifies the DLCI number to be used on the specified subinterface. • **ietf**—(Optional) Specifies IETF encapsulation. This option is required if you did not specify IETF encapsulation in Step 4. **Note** This command includes other options that are not supported when using multipoint bridging.
Step 7	Router(config-fr-dlci)# **bridge-domain** *vlan-id* [**access** | **dot1q** | **dot1q-tunnel**] [**pvst-tlv** *CE-vlan*] [**split-horizon**]	Enables RFC 1490 bridging to map a bridged VLAN to a Frame Relay DLCI, where: • *vlan-id* —Specifies the number of the VLAN to be used in this bridging configuration. The valid range is from 2 to 4094. The VLAN ID must have been previously added to the VLAN database in Step 1. • **access**—(Optional) Enables access-only bridging access mode, in which the bridged connection does not transmit or act upon bridge protocol data unit (BPDU) packets. • **dot1q**—(Optional) Enables IEEE 802.1Q tagging to preserve the class of service (CoS) information from the Ethernet frames across the Frame Relay network. If not specified, the ingress side assumes a CoS value of 0 for QoS purposes. Using the **dot1q** keyword helps avoid misconfiguration because incoming untagged frames, or tagged frames that do not match the specified *vlan-id* are dropped. • **dot1q-tunnel**—(Optional) Enables IEEE 802.1Q tunneling mode, so that service providers can use a single VLAN to support customers who have multiple VLANs, while preserving customer VLAN IDs and keeping traffic in different customer VLANs segregated.

	Command	Purpose
		Note The **access**, **dot1q**, and **dot1q-tunnel** options are mutually exclusive. If you do not specify any of these options, the connection operates in "raw" bridging access mode, which is similar to **access**, except that the connection processes and transmits BPDU packets. • **pvst-tlv** *CE-vlan*—(Optional) When transmitting, translates PVST+ BPDUs into IEEE BPDUs. When receiving, translates IEEE BPDUs into PVST+ BPDUs. *CE-vlan* specifies the customer-edge VLAN in the SSTP Tag-Length-Value (TLV) to be inserted in an IEEE BPDU to a PVST+ BPDU conversion. • **split-horizon**—(Optional) Drops egress traffic going out a VC or interface with split-horizon configured, that arrived on an interface with split-horizon configured.

Verifying MPB for Frame Relay

To display information about the DLCIs that have been configured on Frame Relay interfaces, use the **show frame-relay vlan** command.

```
Router# show frame-relay vlan
Interface           Bridge DLCI     Domain
POS3/1/0.100             100          100
```

Configuring MPB for Gigabit Ethernet

Beginning in Cisco IOS Release 12.2(33)SRA, MPB support is added on the Cisco 7600 SIP-400 to multiplex different VLANs that are configured across multiple Gigabit Ethernet subinterfaces into a single broadcast domain. Gigabit Ethernet interfaces can also reside on different Cisco 7600 SIP-400s and belong to the same bridge domain.

MPB for Gigabit Ethernet Configuration Guidelines

- The Cisco 7600 SIP-400 can support a total of up to 4096 subinterfaces and bridge-domain instances per VLAN. For example, one subinterface with a configured VLAN using MPB will consume two of the available 4096 total allowable subinterfaces and bridge domains combined.
- Up to 60 subinterfaces can be put into the same bridge domain on the Cisco 7600 SIP-400.

To configure MPB for Gigabit Ethernet, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **vlan** {*vlan-id* | *vlan-range*}	Adds the specified VLAN IDs to the VLAN database and enters VLAN configuration mode, where: • *vlan-id*—Specifies a single VLAN ID. The valid range is from 2 to 4094. • *vlan-range*—Specifies multiple VLAN IDs, as either a list or a range. The *vlan-range* can contain a list of the VLAN IDs, separated by a comma (,), dash (-), or both. **Note** Before you can use a VLAN for multipoint bridging, you must manually enter its VLAN ID into the VLAN database.
Step 2	Router(config)# **interface gigabitethernet** *slot***/***subslot***/***port***.***subinterface*	Specifies or creates a Gigabit Ethernet subinterface and enters subinterface configuration mode, where: • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • **.***subinterface*—Specifies the number of the subinterface on the interface port.

	Command	Purpose
Step 3	Router(config-subif) **encapsulation dot1q** *vlan-id*	Enables IEEE 802.1Q encapsulation on the interface, where *vlan-id* specifies the virtual LAN identifier. The allowed range is from 1 to 4095.
Step 4	Router(config-subif)# **bridge-domain** *vlan-id* [**dot1q** | **dot1q-tunnel**] [**bpdu** {**drop** | **transparent**}] [**split-horizon**]	Enables bridging of VLANs across Gigabit Ethernet subinterfaces, where: • *vlan-id* —Specifies the number of the VLAN to be used in this bridging configuration. The valid range is from 2 to 4094. The VLAN ID must have been previously added to the VLAN database in Step 1. • **dot1q**—(Optional) Enables IEEE 802.1Q tagging to preserve the class of service (CoS) information from the Ethernet frames across the ATM network. If not specified, the ingress side assumes a CoS value of 0 for QoS purposes. • **dot1q-tunnel**—(Optional) Enables IEEE 802.1Q tunneling mode, so that service providers can use a single VLAN to support customers who have multiple VLANs, while preserving customer VLAN IDs and keeping traffic in different customer VLANs segregated. **Note** The **dot1q** and **dot1q-tunnel** options are mutually exclusive. If you do not specify either of these options, the connection operates in "raw" bridging access mode, which is similar to access, except that the connection processes and transmits BPDU packets. • **bpdu** {**drop** | **transparent**}**—**(Optional) Specifies whether or not BPDUs are processed or dropped, where: – **drop**—Specifies BPDU packets are dropped on the subinterface. – **transparent**—Specifies BPDU packets are forwarded as data on the subinterface, but not processed. • **split-horizon**—(Optional) Drops egress traffic going out a VC or interface with split-horizon configured, that arrived on an interface with split-horizon configured.

Configuring PPP Bridging Control Protocol Support

The Bridging Control Protocol (BCP) feature on the SIPs and SPAs enables forwarding of Ethernet frames over serial and SONET networks, and provides a high-speed extension of enterprise LAN backbone traffic through a metropolitan area. The implementation of BCP on the SPAs includes support for IEEE 802.1D Spanning Tree Protocol, IEEE 802.1Q Virtual LAN (VLAN), and high-speed switched LANs.

The Bridging Control Protocol (BCP) feature provides support for BCP to Cisco devices, as described in RFC 3518, *Point-to-Point Protocol (PPP) Bridging Control Protocol (BCP)*. The Cisco implementation of BCP is a VLAN infrastructure that does not require the use of subinterfaces to group Ethernet 802.1Q trunks and the corresponding PPP links. This approach enables users to process VLAN encapsulated packets without having to configure subinterfaces for every possible VLAN configuration.

BCP operates in two different modes:

- Trunk mode BCP (**switchport**)—A single BCP link can carry multiple VLANs.
- Single-VLAN BCP (**bridge-domain**)—A single BCP link carries only one VLAN.

In addition, in Cisco IOS Release 12.2(33)SRA, BCP is supported over dMLPPP links on the Cisco 7600 SIP-200 with the 2-Port and 4-Port Channelized T3 SPA and 8-Port Channelized T1/E1 SPA. BCP over dMLPPP is supported in trunk mode only.

BCP Feature Compatibility

Table 4-8 provides information about where the BCP features are supported.

Table 4-8 BCP Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Trunk mode BCP (**switchport**)	In Cisco IOS Release 12.2(18)SXE and later: • 2-Port and 4-Port Channelized T3 SPA • 2-Port and 4-Port Clear Channel T3/E3 SPA • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port OC-3c/STM-1 POS SPA Support for the following SPA was added in Cisco IOS Release 12.2(33)SRA: • 1-Port Channelized OC-3/STM-1 SPA	In Cisco IOS Release 12.2(18)SXE and later: • 1-Port OC-12c/STM-4 POS SPA • 2-Port and 4-Port OC-3c/STM-1 POS SPA • 1-Port OC-48c/STM-16 POS SPA	Not supported.

Table 4-8 BCP Feature Compatibility by SIP and SPA Combination (continued)

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Tag-native Mode for Trunk BCP (**switchport**)	• In Cisco IOS 12.2SX releases—Not supported. • In Cisco IOS Release 12.2(33)SRA: – 2-Port and 4-Port Channelized T3 SPA – 2-Port and 4-Port Clear Channel T3/E3 SPA – 8-Port Channelized T1/E1 SPA – 2-Port and 4-Port OC-3c/STM-1 POS SPA – 1-Port Channelized OC-3/STM-1 SPA	• In Cisco IOS 12.2SX releases—Not supported. • In Cisco IOS Release 12.2(33)SRA: – 1-Port OC-12c/STM-4 POS SPA – 2-Port and 4-Port OC-3c/STM-1 POS SPA – 1-Port OC-48c/STM-16 POS SPA	Not supported.
Single-VLAN BCP (**bridge-domain**)	In Cisco IOS Release 12.2(18)SXE and later: • 2-Port and 4-Port Channelized T3 SPA • 2-Port and 4-Port Clear Channel T3/E3 SPA • 8-Port Channelized T1/E1 SPA • 2-Port and 4-Port OC-3c/STM-1 POS SPA Support for the following SPA was added in In Cisco IOS Release 12.2(33)SRA: • 1-Port Channelized OC-3/STM-1 SPA	In Cisco IOS Release 12.2(33)SRA: • 1-Port OC-12c/STM-4 POS SPA • 2-Port and 4-Port OC-3c/STM-1 POS SPA • 1-Port OC-48c/STM-16 POS SPA	Not supported.
BCP over dMLPPP (trunk mode only)	In Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port Channelized T3 SPA • 8-Port Channelized T1/E1 SPA	Not supported.	Not supported.

BCP Configuration Guidelines

When configuring BCP support for SPAs on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, consider the following guidelines:

- Be sure to refer to Table 4-8 for feature compatibility information.
- Beginning in Cisco IOS Release 12.2(33)SRA, QoS is supported on bridged interfaces. In Cisco IOS Release 12.2(18)SXF2 and earlier, QoS is not supported on bridged interfaces.
- Although RFC 3518 specifies support for Token Ring and Fiber Distributed Data Interface (FDDI), BCP on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400 supports only Ethernet currently.

Configuring BCP in Trunk Mode

When BCP is configured in trunk mode, a single BCP link can carry multiple VLANs. This usage of BCP is consistent with that of normal Ethernet trunk ports.

Trunk Mode BCP Configuration Guidelines

When configuring BCP support in trunk mode for SPAs on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, consider the following guidelines:

- Be sure to refer to Table 4-8 for feature compatibility information.
- There are some differences between the Ethernet trunk ports and BCP trunk ports.
  - Ethernet trunk ports support ISL and 802.1Q encapsulation, but BCP trunk ports support only 802.1Q.
  - Ethernet trunk ports support Dynamic Trunk Protocol (DTP), which is used to automatically determine the trunking status of the link. BCP trunk ports are always in trunk state and no DTP negotiation is performed.
  - The default behavior of Ethernet trunk ports is to allow all VLANs on the trunk. The default behavior of BCP trunks is to disallow all VLANs. This means that VLANs that need to be allowed have to be explicitly configured on the BCP trunk port.
- Use the **switchport** command under the WAN interface when configuring trunk mode BCP.
- The SIPs support the following maximum number of BCP ports on any given VLAN:
  - In Cisco IOS Release 12.2(18)SXE and later—Maximum of 60 BCP ports
  - In Cisco IOS Release 12.2(33)SRA—Maximum of 112 BCP ports on Cisco 7600 SIP-200 and maximum of 120 BCP ports on Cisco 7600 SIP-400.
- To use VLANs in trunk mode BCP, you must use the **vlan** command to manually add the VLANs to the VLAN database. The default behavior for trunk mode BCP allows no VLANs.
- Trunk mode BCP is not supported on VLAN IDs 0, 1006–1023, and 1025.
- The native VLAN (VLAN1) has the following restrictions for trunk mode BCP:
  - In Cisco IOS Release 12.2SX—The native VLAN is not supported.
  - Beginning in Cisco IOS Release 12.2(33)SRA—The native VLAN is supported.
- For trunk mode BCP (switchport), STP interoperability is the same as that of Ethernet switchports. This means that the STP path cost of WAN links can be changed and other STP functionality such as BPDU Guard and PortFast will work on the WAN links. However, it is not recommended to change the default values.
- VLAN Trunking Protocol (VTP) is supported.

Note The management VLAN, VLAN 1, must be explicitly enabled on the trunk to send VTP advertisements.

To configure BCP in trunk mode, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **vlan dot1q tag native**	(Optional) Enables dot1q tagging for all VLANs in a trunk. By default, packets on the native VLAN are sent untagged. When you enable dot1q tagging, packets are tagged with the native VLAN ID.
Step 2	**1-Port Channelized OC-3/STM-1 SPA** Router(config)# **interface serial** *address* **2-Port and 4-Port Channelized T3 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***/***t1-number***:***channel-group* **8-Port Channelized T1/E1 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***:***channel-group* **1-Port OC-12c/STM-4 POS SPA or 2-Port and 4-Port OC-3c/STM-1 POS SPA** Router(config)# **interface pos** *slot***/***subslot***/***port*	Specifies an interface and enters interface configuration mode, where: • *address*—For the different supported syntax options for the *address* argument for the 1-Port Channelized OC-3/STM-1 SPA, refer to the "Interface Naming" section of the "Configuring the 1-Port Channelized OC-3/STM-1 SPA" chapter. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • *t1-number*—Specifies the logical T1 number in channelized mode. • *channel-group*—Specifies the logical channel group assigned to the time slots within the T1 or E1 group.
Step 3	Router(config-if)# **switchport**	Puts an interface that is in Layer 3 mode into Layer 2 mode for Layer 2 configuration. PPP encapsulation is automatically configured, and the interface is automatically configured for trunk mode and nonegotiate status.
Step 4	Router(config-if)# **shutdown**	Disables the interface.

	Command	Purpose
Step 5	Router(config-if)# **no shutdown**	Restarts the disabled interface.
Step 6	Router(config-if)# **switchport trunk allowed vlan** {**all** | {**add** | **remove** | **except**} *vlan-list* [,*vlan-list...*] | *vlan-list* [,*vlan-list...*]}	(Optional) Controls which VLANs can receive and transmit traffic on the trunk, where: • **all**—Enables all applicable VLANs. • **add** *vlan-list* [**,***vlan-list...*]—Appends the specified list of VLANs to those currently set instead of replacing the list. • **remove** *vlan-list* [**,***vlan-list...*]—Removes the specified list of VLANs from those currently set instead of replacing the list. • **except** *vlan-list* [**,***vlan-list...*]—Excludes the specified list of VLANs from those currently set instead of replacing the list. • *vlan-list* [**,***vlan-list...*]—Specifies a single VLAN number from 1 to 4094, or a continuous range of VLANs that are described by two VLAN numbers from 1 to 4094. You can specify multiple VLAN numbers or ranges using a comma-separated list. To specify a range of VLANs, enter the smaller VLAN number first, separated by a hyphen and the larger VLAN number at the end of the range. **Note** Do not enable the reserved VLAN range (1006 to 1024) on trunks when connecting a Cisco 7600 series router running the Cisco IOS software on both the supervisor engine and the MSFC to a Cisco 7600 series router running the Catalyst operating system. These VLANs are reserved in Cisco 7600 series routers running the Catalyst operating system. If enabled, Cisco 7600 series routers running the Catalyst operating system may error-disable the ports if there is a trunking channel between these systems.

Verifying BCP in Trunk Mode

Because the PPP link has to flap (be brought down and renegotiated), it is important that you run the following **show** commands after you configure BCP in trunk mode to confirm the configuration:

Command	Purpose
1-Port Channelized OC-3/STM-1 SPA Router# **show interfaces** [**serial** *address*] **trunk** [**module** *number*] **2-Port and 4-Port Channelized T3 SPA** Router# **show interfaces** [**serial** *slot*/*subslot*/*port*/*t1-number*:*channel-group*] **trunk** [**module** *number*] **8-Port Channelized T1/E1 SPA** Router# **show interfaces** [**serial** *slot*/*subslot*/*port*:*channel-group*] **trunk** [**module** *number*] **1-Port OC-12c/STM-4 POS SPA or 2-Port and 4-Port OC-3c/STM-1 POS SPA** Router# **show interfaces** [**pos** *slot*/*subslot*/*port*] **trunk** [**module** *number*]	Displays the interface-trunk information, where: • *address*—For the different supported syntax options for the *address* argument for the 1-Port Channelized OC-3/STM-1 SPA, refer to the “Interface Naming” section of the “Configuring the 1-Port Channelized OC-3/STM-1 SPA” chapter. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • *t1-number*—Specifies the logical T1 number in channelized mode. • *channel-group*—Specifies the logical channel group assigned to the time slots within the T1 or E1 group. • **module** *number*—(Optional) Specifies the chassis slot number of the SIP and displays information for all interfaces of the SPAs in that SIP.

Command	Purpose
1-Port Channelized OC-3/STM-1 SPA Router# **show interfaces** [**serial** *address*] **switchport** [**module** *number*] **2-Port and 4-Port Channelized T3 SPA** Router# **show interfaces** [**serial** *slot***/***subslot***/***port***/***t1-number***:***channel-group*] **switchport** [**module** *number*] **8-Port Channelized T1/E1 SPA** Router# **show interfaces** [**serial** *slot***/***subslot***/***port***:***channel-group*] **switchport** [**module** *number*] **1-Port OC-12c/STM-4 POS SPA or 2-Port and 4-Port OC-3c/STM-1 POS SPA** Router# **show interfaces** [**pos** *slot***/***subslot***/***port*] **switchport** [**module** *number*]	Displays the administrative and operational status of a switching (nonrouting) port, where: • *address*—For the different supported syntax options for the *address* argument for the 1-Port Channelized OC-3/STM-1 SPA, refer to the "Interface Naming" section of the "Configuring the 1-Port Channelized OC-3/STM-1 SPA" chapter. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • *t1-number*—Specifies the logical T1 number in channelized mode. • *channel-group*—Specifies the logical channel group assigned to the time slots within the T1 or E1 group. • **module** *number*—(Optional) Specifies the chassis slot number of the SIP and displays information for all interfaces of the SPAs in that SIP.

The following output of the **show interfaces** commands provide an example of the information that is displayed when BCP is configured in trunk mode.

Note When switchport is configured, the encapsulation is automatically changed to PPP.

```
Router# show interfaces trunk
Port      Mode         Encapsulation  Status        Native vlan
PO4/1/0   on           802.1q         trunking      1

Port      Vlans allowed on trunk
PO4/1/0   1-1005,1025-1026,1028-4094

Port      Vlans allowed and active in management domain
PO4/1/0   1,100,200

Port      Vlans in spanning tree forwarding state and not pruned
PO4/1/0   1,100,200

Router# show interfaces switchport

Name: PO4/1/0
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
```

```
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan: none
Trunking VLANs Enabled: 100
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Unknown unicast blocked: disabled
Unknown multicast blocked: disabled

Router# show interfaces pos4/1/0
POS4/1/0 is up, line protocol is up
  Hardware is Packet over Sonet
  MTU 4470 bytes, BW 155000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, crc 16, loopback not set
  Keepalive set (10 sec)
  Scramble disabled
  LCP Open
  Open: BRIDGECP, CDPCP
  Last input 00:00:05, output 00:00:05, output hang never
  Last clearing of "show interface" counters 18:48:09
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 1000 bits/sec, 1 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    13161719 packets input, 1145463122 bytes, 0 no buffer
    Received 0 broadcasts (0 IP multicast)
    0 runts, 0 giants, 0 throttles
             0 parity
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    1685 packets output, 620530 bytes, 0 underruns
    0 output errors, 0 applique, 30 interface resets
    0 output buffer failures, 0 output buffers swapped out
    11 carrier transitions
```

Configuring BCP in Single-VLAN Mode

When BCP is configured in single-VLAN mode, a single BCP link carries only one VLAN. This is considered BCP in access mode.

Single-VLAN Mode BCP Configuration Guidelines

When configuring BCP support in single-VLAN mode for SPAs on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, consider the following guidelines:

- Be sure to refer to Table 4-8 for feature compatibility information.
- Use the **bridge-domain** *vlan-id* **dot1q** form of the command under a WAN interface or an ATM PVC. The **dot1q** keyword is necessary. It indicates that all frames on the BCP link will be tagged with a 802.1Q header. Untagged frames received on a BCP link will be dropped.

- For serial and POS SPA interfaces, the encapsulation of the interface must be PPP; otherwise, the **bridge-domain** command will not be accepted.
- The ATM SPAs on the Cisco 7600 series router do not support single-VLAN BCP.
- For single-VLAN BCP, you can configure the following maximum number of VCs per VLAN:
  - In Cisco IOS Release 12.2SX—60 VCs or interfaces per VLAN per chassis.
  - Beginning in Cisco IOS Release 12.2(33)SRA—112 VCs or interfaces per VLAN per Cisco 7600 SIP-200; 120 VCs or interfaces per VLAN per Cisco 7600 SIP-400.
- VLANs must be manually added to the VLAN database, using the **vlan** command, to be able to use those VLANs in single-VLAN BCP.
- BCP is not supported on VLAN IDs 0, 1 (native), 1006–1023, and 1025.
- For single-VLAN BCP, only basic Spanning Tree Protocol (STP) interoperability is supported. This means that single-VLAN BCP interfaces will participate in the STP domain and the correct path cost of the links will be calculated; however, changing any STP parameters for the link is not supported.
- VLAN Trunking Protocol (VTP) is not supported on single-VLAN BCP.

To configure BCP in single-VLAN mode on serial or POS SPAs, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	**1-Port Channelized OC-3/STM-1 SPA** Router(config)# **interface serial** *address* **2-Port and 4-Port Channelized T3 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***/***t1-number***:***channel-group* **8-Port Channelized T1/E1 SPA** Router(config)# **interface serial** *slot***/***subslot***/***port***:***channel-group* **1-Port OC-12c/STM-4 POS SPA or 2-Port and 4-Port OC-3c/STM-1 POS SPA** Router(config)# **interface pos** *slot***/***subslot***/***port*	Specifies an interface and enters interface configuration mode, where: • *address*—For the different supported syntax options for the *address* argument for the 1-Port Channelized OC-3/STM-1 SPA, refer to the "Interface Naming" section of the "Configuring the 1-Port Channelized OC-3/STM-1 SPA" chapter. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. • *t1-number*—Specifies the logical T1 number in channelized mode. • *channel-group*—Specifies the logical channel group assigned to the time slots within the T1 or E1 group.
Step 2	Router(config-if)# **no ip address**	Disables IP processing on a particular interface by removing its IP address.
Step 3	Router(config-if)# **encapsulation ppp**	Configures the interface for PPP encapsulation.

	Command	Purpose
Step 4	Router(config-if)# **bridge-domain** *vlan-id* [**dot1q** | **dot1q-tunnel**]	Establishes a domain and tags all Ethernet frames on the BCP link with the 802.1Q header, where: • *vlan-id*—Specifies the number of the VLAN to be used in this bridging configuration. The valid range is from 2 to 4094. The VLAN ID must have been previously added to the VLAN database. • **dot1q**—(Optional) Enables IEEE 802.1Q tagging to preserve the class of service (CoS) information from the Ethernet frames across the WAN interface. If not specified, the ingress side assumes a CoS value of 0 for QoS purposes. Using the **dot1q** keyword helps avoid misconfiguration because incoming untagged frames, or tagged frames that do not match the specified *vlan-id* are dropped. • **dot1q-tunnel**—(Optional) Enables IEEE 802.1Q tunneling mode, so that service providers can use a single VLAN to support customers who have multiple VLANs, while preserving customer VLAN IDs and keeping traffic in different customer VLANs segregated.
Step 5	Router(config-if)# **shutdown**	Disables the interface.
Step 6	Router(config-if)# **no shutdown**	Restarts the disabled interface.

Verifying BCP in Single-VLAN Mode

Because the PPP link has to flap (be brought down and renegotiated), it is important that you run the following **show** command after you configure BCP in single-VLAN mode to confirm the configuration:

```
Router# show interfaces pos4/1/0
POS4/1/0 is up, line protocol is up
  Hardware is Packet over Sonet
  MTU 4470 bytes, BW 155000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, crc 16, loopback not set
  Keepalive set (10 sec)
  Scramble disabled
  LCP Open
  Open: BRIDGECP, CDPCP
  Last input 00:00:09, output 00:00:09, output hang never
  Last clearing of "show interface" counters 00:00:24
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 1000 bits/sec, 1 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     32 packets input, 1709 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     17 packets output, 1764 bytes, 0 underruns
     0 output errors, 0 applique, 3 interface resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions
```

Configuring BCP over dMLPPP

Beginning in Cisco IOS Release 12.2(33)SRA, BCP is supported over dMLPPP links on the Cisco 7600 SIP-200 with the 2-Port and 4-Port Channelized T3 SPA and 8-Port Channelized T1/E1 SPA. BCP over dMLPPP is supported in trunk mode only.

For more information about configuring the BCP over dMLPPP feature, see Chapter 17, "Configuring the 8-Port Channelized T1/E1 SPA," and Chapter 19, "Configuring the 2-Port and 4-Port Channelized T3 SPAs."

Configuring Virtual Private LAN Service

Virtual Private LAN Service (VPLS) enables geographically separate LAN segments to be interconnected as a single bridged domain over a packet switched network, such as IP, MPLS, or a hybrid of both.

VPLS solves the network reconfiguration problems at the CE that are associated with Layer 2 Virtual Private Network (L2VPN) implementations. The current Cisco IOS software L2VPN implementation builds a point-to-point connection to interconnect the two attachment VCs of two peering customer sites. To communicate directly among all sites of an L2VPN network, a distinct emulated VC needs to be created between each pair of peering attachment VCs. For example, when two sites of the same L2VPN network are connected to the same PE, it requires that two separate emulated VCs be established towards a given remote site, instead of sharing a common emulated VC between these two sites. For a L2VPN customer who uses the service provider backbone to interconnect its LAN segments, the current implementation effectively turns its multiaccess broadcast network into a fully meshed point-to-point network, which requires extensive reconfiguration on the existing CE devices.

VPLS is a multipoint L2VPN architecture that connects two or more customer devices using EoMPLS bridging techniques. VPLS with EoMPLS uses an MPLS-based provider core, where the PE routers have to cooperate to forward customer Ethernet traffic for a given VPLS instance in the core.

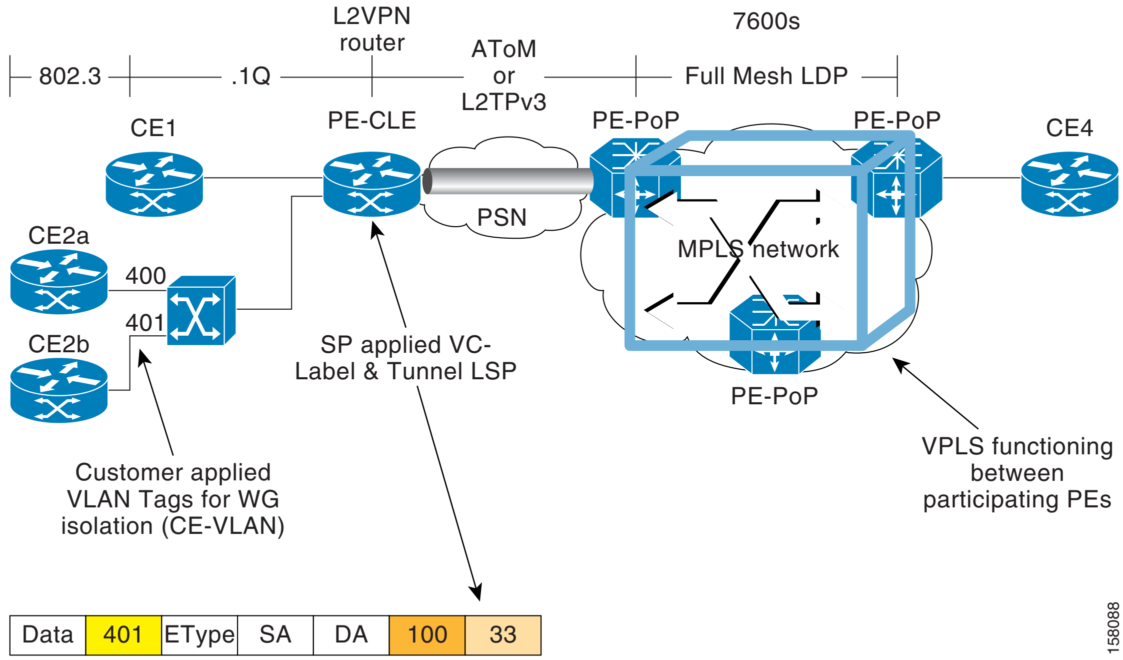
VPLS uses the provider core to join multiple attachment circuits together to simulate a virtual bridge that connects the multiple attachment circuits together. From a customer point of view, there is no topology for VPLS. All of the CE devices appear to connect to a logical bridge emulated by the provider core.

Hierarchical Virtual Private LAN Service with MPLS to the Edge

In a flat or non-hierarchical VPLS configuration, a full mesh of pseudowires (PWs) is needed between all PE nodes. A *pseudowire* defines a VLAN and its corresponding pseudoport.

Hierarchical Virtual Private LAN Service (H-VPLS) reduces both signaling and replication overhead by using a combination of full-mesh and hub-and-spoke configurations. Hub-and-spoke configurations operate with split horizon to allow packets to be switched between pseudowires (PWs), which effectively reduce the number of PWs between PEs.

Figure 4-3 H-VPLS with MPLS to the Edge Network



In the H-VPLS with MPLS to the edge architecture, Ethernet Access Islands (EAIs) work in combination with a VPLS core network, with MPLS as the underlying transport mechanism. EAIs operate like standard Ethernet networks. In Figure 4-3, devices CE1, CE2a and CE2b reside in an EAI. Traffic from any CE devices within the EAI are switched locally within the EAI by the user-facing provider edge (UPE) device along the computed spanning-tree path. Each user-facing provider edge device is connected to one or more network-facing provider edge devices using PWs. The traffic local to the UPE is not forward to any network-facing provider edge devices.

VPLS Configuration Guidelines

When configuring VPLS on a SIP, consider the following guidelines:

- For support of specific VPLS features by SIP, see Table 4-9.
- The SIPs support up to 4000 VPLS domains per Cisco 7600 series router.
- The SIPs support up to 60 VPLS peers per domain per Cisco 7600 series router.
- The SIPs support up to 30,000 pseudowires, used in any combination of domains and peers up to the 4000-domain or 60-peer maximums. For example, support of up to 4000 domains with 7 peers, or up to 60 peers in 500 domains.
- When configuring VPLS on a Cisco 7600 SIP-600, consider the following guidelines:
  - Q-in-Q (the ability to map a single 802.1Q tag or a random double tag combination into a VPLS instance, a Layer 3 MPLS VPN, or an EoMPLS VC) is not supported.
  - H-VPLS with Q-in-Q edge—Requires a Cisco 7600 SIP-600 in the uplink, and any LAN port or Cisco 7600 SIP-600 on the downlink.
- H-VPLS with MPLS edge requires either an OSM module, Cisco 7600 SIP-600, or Cisco 7600 SIP-400 in both the downlink (facing UPE) and uplink (MPLS core).
- The Cisco 7600 SIP-400 and Cisco 7600 SIP-600 provide Transparent LAN Services (TLS) and Ethernet Virtual Connection Services (EVCS).

- The Cisco 7600 SIP-400 does not support redundant PW links from a UPE to multiple NPEs.
- For information about configuring VPLS on the SIPs, consider the guidelines in this document and then refer to the "Virtual Private LAN Services on the Optical Services Modules" section of the *Optical Services Module Software Configuration Note* for the Cisco 7600 series router at the following URL:

  http://www.cisco.com/application/pdf/en/US/guest/products/ps368/c1069/ccmigration_09186a008069bfcc.pdf

VPLS Feature Compatibility

Table 4-9 provides information about where the VPLS features are supported.

Table 4-9 VPLS Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
H-VPLS with MPLS edge	Not supported.	In Cisco IOS Release 12.2(33)SRA: • 2-Port Gigabit Ethernet SPA • 2-Port and 4-Port OC-3c/STM-1 POS SPA • 1-Port OC-12c/STM-4 POS SPA • 1-Port OC-48c/STM-16 POS SPA	In Cisco IOS Release 12.2(18)SXF and later: • 1-Port 10-Gigabit Ethernet SPA • 5-Port Gigabit Ethernet SPA • 10-Port Gigabit Ethernet SPA • 1-Port OC-192c/STM-64 POS/RPR SPA • 2-Port and 4-Port OC-48c/STM-16 POS SPA Support for the following SPAs was added in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-48c/STM-16 POS SPA

Table 4-9 VPLS Feature Compatibility by SIP and SPA Combination (continued)

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
H-VPLS with Q-in-Q edge	Not supported.	Not supported.	In Cisco IOS Release 12.2(18)SXF and later: • 1-Port 10-Gigabit Ethernet SPA • 5-Port Gigabit Ethernet SPA • 10-Port Gigabit Ethernet SPA • 1-Port OC-192c/STM-64 POS/RPR SPA • 2-Port and 4-Port OC-48c/STM-16 POS SPA Support for the following SPAs was added in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-48c/STM-16 POS SPA
VPLS with point-to-multipoint EoMPLS and fully-meshed PE configuration	Not supported.	In Cisco IOS Release 12.2(33)SRA: • 2-Port Gigabit Ethernet SPA • 2-Port and 4-Port OC-3c/STM-1 POS SPA • 1-Port OC-12c/STM-4 POS SPA • 1-Port OC-48c/STM-16 POS SPA	In Cisco IOS Release 12.2(18)SXF and later: • 1-Port 10-Gigabit Ethernet SPA • 5-Port Gigabit Ethernet SPA • 10-Port Gigabit Ethernet SPA • 1-Port OC-192c/STM-64 POS/RPR SPA • 2-Port and 4-Port OC-48c/STM-16 POS SPA Support for the following SPAs was added in Cisco IOS Release 12.2(33)SRA: • 2-Port and 4-Port OC-48c/STM-16 POS SPA

Configuring MPLS Features on a SIP

Many of the MPLS features supported on the FlexWAN and Enhanced FlexWAN modules on the Cisco 7600 series router are also supported by the SIPs. For a list of the supported MPLS features on the SIPs, see Chapter 3, "Overview of the SIPs and SSC."

This section describes those MPLS features that have SIP-specific configuration guidelines. After you review the SIP-specific guidelines described in this document, then refer to the following URL for more information about configuring MPLS features:

http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/cfgnotes/flexport/combo/flexmpls.htm

This section includes the following topics:

- Configuring Any Transport over MPLS on a SIP, page 4-51
- Configuring Hierarchical Virtual Private LAN Service (H-VPLS) with MPLS to the Edge, page 4-54
- Configuring MPLS Traffic Engineering Class-Based Tunnel Selection (CBTS) on the Cisco 7600 SIP-600, page 4-54

Configuring Any Transport over MPLS on a SIP

Any Transport over MPLS (AToM) transports Layer 2 packets over a Multiprotocol Label Switching (MPLS) backbone. AToM uses a directed Label Distribution Protocol (LDP) session between edge routers for setting up and maintaining connections. Forwarding occurs through the use of two levels of labels, switching between the edge routers. The external label (tunnel label) routes the packet over the MPLS backbone to the egress Provider Edge (PE) at the ingress PE. The VC label is a demuxing label that determines the connection at the tunnel endpoint (the particular egress interface on the egress PE as well as the virtual path identifier [VPI]/virtual channel identifier [VCI] value for an ATM Adaptation Layer 5 [AAL5] protocol data unit [PDU], the data-link connection identifier [DLCI] value for a Frame Relay PDU, or the virtual LAN [VLAN] identifier for an Ethernet frame).

For specific information about configuring AToM features, refer to the *FlexWAN and Enhanced FlexWAN Module Installation and Configuration Note* located at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/cfgnotes/flexport/combo/flexmpls.htm

Note When referring to the FlexWAN documentation, be sure to note any SIP-specific configuration guidelines described in this document.

Cisco 7600 SIP-200 AToM Features

The Cisco 7600 SIP-200 supports the following AToM features:

- ATM over MPLS (ATMoMPLS)—AAL5 VC mode
- Ethernet over MPLS (EoMPLS)—(Single cell relay) VC mode
- Frame Relay over MPLS (FRoMPLS)
- FRoMPLS with dMLFR—Supported between the CE and PE devices.
- High-Level Data Link Control (HDLC) over MPLS (HDLCoMPLS)
- PPP over MPLS (PPPoMPLS)—Not supported with dMLPPP or dLFI
- Hierarchical QoS for EoMPLS VCs

Cisco 7600 SIP-200 AToM Configuration Guidelines

When configuring AToM with a Cisco 7600 SIP-200, consider the following guidelines:

- You cannot use a SIP-200 and an Ethernet SPA on the customer-facing side because the Ethernet SPA is a Layer 3 only interface.

- Because the SIP-200 supports WAN interfaces, you can use the SIP-200 for non-Ethernet access (FR,HDLC,ATM,PPP) at the customer-facing side.
- For VLAN-based xconnect (also called linecard-based EoMPLS), the customer-facing port must be a Layer 2 port and the backbone-facing card must be a Layer 3 port.
- The SIP-200 does not supportdot1q subinterface-based xconnect towards the edge.

Cisco 7600 SIP-400 AToM Features

The Cisco 7600 SIP-400 supports the following AToM features:

- ATMoMPLS—AAL0 mode (single cell relay only)
- ATMoMPLS—AAL5 mode
- EoMPLS—Port mode
- EoMPLS—VLAN mode
- FRoMPLS—DLCI mode
- Beginning in Cisco IOS Release 12.2(33)SRA:
  - Hierarchical QoS for EoMPLS VCs
  - HDLCoMPLS
  - PPPoMPLS
  - ATM local switching

Cisco 7600 SIP-400 AToM Configuration Guidelines

When configuring AToM with a Cisco 7600 SIP-400, consider the following guidelines:

- The Cisco 7600 SIP-400 is not supported with a Supervisor Engine 1, Supervisor Engine 1A, Supervisor Engine 2, or Supervisor Engine 720 PFC3A.
- The Cisco 7600 SIP-400 is not supported with PFC-2-based systems.
- For AToM in Cisco IOS 12.2SX releases, the Cisco 7600 SIP-400 does not support the following features when they are located in the data path. This means you should not configure the following features if the SIP is facing the customer edge (CE) or the MPLS core:
  - HDLCoMPLS
  - PPPoMPLS
  - VPLS
- For AToM beginning in Cisco IOS Release 12.2(33)SRA, the Cisco 7600 SIP-400 supports the following features on CE-facing interfaces:
  - HDLCoMPLS
  - PPPoMPLS
  - VPLS
- The Cisco 7600 SIP-400 supports EoMPLS with directly connected provider edge (PE) devices when the Cisco 7600 SIP-400 is on the MPLS core side of the network.
- The Cisco 7600 SIP-400 does not support the ability to enable or disable tunneling of Layer 2 packets, such as for the VLAN Trunking Protocol (VTP), Cisco Discovery Protocol (CDP), and bridge protocol data unit (BPDU). The Cisco 7600 SIP-400 tunnels BPDUs, and always blocks VTP and CDP packets from the tunnel.

- In ATMoMPLS AAL5 and cell mode, the Cisco 7600 SIP-400 supports non-matching VPIs/VCIs between PEs if the Cisco 7600 SIP-400 is on both sides of the network.
- The Cisco 7600 SIP-400 supports matching on FR-DE to set MPLS-EXP for FRoMPLS.
- The Cisco 7600 SIP-400 supports use of the **xconnect** command to configure AToM circuits for all AToM connection types except ATMoMPLS. For ATMoMPLS, you must use the **mpls l2 transport route** command.

  For information about configuring the **xconnect** command for AToM circuits, refer to the MPLS examples using the **xconnect** command at the following URL:

  http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/cfgnotes/optical/122sx/mpls.htm
- The Cisco 7600 SIP-400 does not support the following QoS classification features with AToM:
  - Matching on data-link connection identifier (DLCI) is unsupported.
  - Matching on virtual LAN (VLAN) is unsupported.
  - Matching on class of service (CoS) is unsupported in Cisco IOS Release 12.2(18)SXE and Cisco IOS Release 12.2(18)SXE2 only. Beginning in Cisco IOS Release 12.2(18)SXF, it is supported with the 2-Port Gigabit Ethernet SPA.
  - Matching on input interface is unsupported.
  - Matching on packet length is unsupported.
  - Matching on media access control (MAC) address is unsupported.
  - Matching on protocol type, including Border Gateway Protocol (BGP), is unsupported.

Understanding MPLS Imposition on the Cisco 7600 SIP-400 to Set MPLS Experimental Bits

The MPLS imposition function encapsulates non-MPLS frames (such as Ethernet, VLAN, Frame Relay, ATM, or IP) into MPLS frames. MPLS disposition performs the reverse function.

An input QoS policy map is applied to ingress packets *before* MPLS imposition takes place. This means that the packets are treated as non-MPLS frames, so any MPLS-related matches have no effect. In the case of marking experimental (EXP) bits using the **set mpls experimental** command, the information is passed to the AToM or MPLS component to set the EXP bits. After imposition takes place, the frame becomes an MPLS frame and an output QoS policy map (if it exists) can apply MPLS-related criteria.

On the egress side, an output QoS policy map is applied to the egress packets *after* MPLS disposition takes place. This means that packets are treated as non-MPLS frames, so any MPLS-related criteria has no effect. Before disposition, the frame is an MPLS frame and the input QoS policy map (if it exists) can apply MPLS-related criteria.

The Encoded Address Recognition Logic (EARL) is a centralized processing engine for learning and forwarding packets based upon MAC address on the Cisco 7600 series router supervisor engines. The EARL stores the VLAN, MAC address, and port relationships. These relationships are used to make switching decisions in hardware. The EARL engine also performs MPLS imposition, and the MPLS EXP bits are copied either from the IP TOS field (using **trust dscp** or **trust precedence** mode), or from the DBUS header QoS field (using **trust cos** mode).

When using the 2-Port Gigabit Ethernet SPA with the Cisco 7600 SIP-400 as the customer-side interface configured for 802.1Q encapsulation for IP imposition with MPLS, the Layer 2 CoS value is not automatically copied into the corresponding MPLS packet’s EXP bits. Instead, the value in the IP precedence bits is copied.

To maintain the 802.1Q CoS values, classify the imposition traffic on the customer-facing Gigabit Ethernet interface in the input direction to match on CoS value, and then set the MPLS experimental action for that class as shown in the following example:

```
Router(config)# class-map cos0
Router(config-cmap)# match cos 0
Router(config-cmap)# exit
!
Router(config)# class-map cos1
Router(config-cmap)# match cos 1
Router(config-cmap)# exit
!
Router(config)# policy-map policy1
Router(config-pmap)# class cos0
Router(config-pmap-c)# set mpls experimental imposition 0
Router(config-pmap-c)# exit
Router(config-pmap)# class cos1
Router(config-pmap-c)# set mpls experimental imposition 1
```

Cisco 7600 SIP-600 AToM Features

The Cisco 7600 SIP-600 supports the following AToM features:

- Any Transport over MPLS (AToM) support—EoMPLS only (Encoded Address Recognition Logic [EARL]-based and SIP-based EoMPLS)

Configuring Hierarchical Virtual Private LAN Service (H-VPLS) with MPLS to the Edge

The Cisco 7600 SIP-400 and Cisco 7600 SIP-600 support the H-VPLS with MPLS to the Edge feature. For more information about VPLS support on the SIPs, see the "Configuring Virtual Private LAN Service" section on page 4-46.

Configuring MPLS Traffic Engineering Class-Based Tunnel Selection (CBTS) on the Cisco 7600 SIP-600

Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Class-Based Tunnel Selection (CBTS) enables you to dynamically route and forward traffic with different class of service (CoS) values onto different TE tunnels between the same tunnel headend and the same tailend. The TE tunnels can be regular TE or DiffServ-aware TE (DS-TE) tunnels.

The set of TE (or DS-TE) tunnels from the same headend to the same tailend that you configure to carry different CoS values is referred to as a "tunnel bundle." Tunnels are "bundled" by creating a master tunnel and then attaching member tunnels to the master tunnel. After configuration, CBTS dynamically routes and forwards each packet into the tunnel that meets the following requirements:

- Is configured to carry the CoS of the packet
- Has the right tailend for the destination of the packet

Because CBTS offers dynamic routing over DS-TE tunnels and requires minimum configuration, it greatly eases deployment of DS-TE in large-scale networks.

CBTS can distribute all CoS values on eight different tunnels.

CBTS also allows the TE tunnels of a tunnel bundle to exit headend routers through different interfaces.

CTBS configuration involves performing the following tasks:

- Creating multiple (DS-) TE tunnels withe same headend and tailend and indicating on each of these tunnels which COSs are to be transported on the tunnel.
- Creating a master tunnel, attaching the member tunnels to it, and making the master tunnel visible for routing.

MPLS Traffic Engineering Class-Based Tunnel Selection (CBTS) Configuration Guidelines

When configuring MPLS Traffic Engineering Class-Based Tunnel Selection (CBTS), consider the following guidelines:

- CBTS has the following prerequisites:
  - MPLS enabled on all tunnel interfaces
  - Cisco Express Forwarding (CEF) or distributed CEF (dCEF) enabled in general configuration mode
- CBTS has the following restrictions:
  - For a given destination, all CoS values are carried in tunnels terminating at the same tailend. Either all CoS values are carried in tunnels or no values are carried in tunnels. In other words, for a given destination, you cannot map some CoS values in a DS-TE tunnel and other CoS values in a Shortest Path First (SPF) Label Distribution Protocol (LDP) or SPF IP path.
  - No LSP is established for the master tunnel and regular traffic engineering attributes (bandwidth, path option, fast reroute) are irrelevant on a master tunnel. TE attributes (bandwidth, bandwidth pool, preemption, priorities, path options, and so on) are configured completely independently for each tunnel.
  - CBTS does not allow load-balancing of a given EXP value in multiple tunnels. If two or more tunnels are configured to carry a given experimental (EXP) value, CBTS picks one of these tunnels to carry this EXP value.
  - CBTS supports aggregate control of bumping (that is, it is possible to define default tunnels to be used if other tunnels go down. However, CBTS does not allow control of bumping if the default tunnel goes down. CBTS does not support finer-grain control of bumping. For example, if the voice tunnel goes down, redirect voice to T2, but if video goes down, redirect to T3.
  - The operation of CBTS is not supported with Any Transport over MPLS (AToM), MPLS TE Automesh, or label-controlled (LC)-ATM.

Creating Multiple MPLS Member TE or DS-TE Tunnels from the Same Headend to the Same Tailend

Perform the following task to create multiple MPLS member TE or DS-TE tunnels with the same headend and same tailend and to configure EXP values to be carried by each of these tunnels. The procedure begins in global configuration mode.

	Command	Purpose
Step 1	Router(config)# **interface tunnel** *number*	Configures a tunnel interface type and enters interface configuration mode. • *number*—Number of the tunnel interface that you want to create or configure.
Step 2	Router(config-if)# **ip unnumbered** *type number*	Enables IP processing on an interface without assigning an explicit IP address to the interface. • *type*—Type of another interface on which the router has an assigned IP address. • *number*—Number of another interface on which the router has an assigned IP address. It cannot be another unnumbered interface.

	Command	Purpose
Step 3	Router(config-if)# **tunnel destination** {*hostname* | *ip-address*}	Specifies the destination of the tunnel for this path option. • *hostname*—Name of the host destination. • *ip-address*—IP address of the host destination expressed in four-part, dotted decimal notation.
Step 4	Router(config-if)# **tunnel mode mpls traffic-eng**	Sets the mode of a tunnel to MPLS for TE.
Step 5	Router(config-if)# **tunnel mpls traffic-eng bandwidth** [**sub-pool** | **global**] *bandwidth*	Configures the bandwidth for the MPLS TE tunnel. If automatic bandwidth is configured for the tunnel, use the **tunnel mpls traffic-eng bandwidth** command to configure the initial tunnel bandwidth, which is adjusted by the auto-bandwidth mechanism. • **sub-pool**—(Optional) Indicates a subpool tunnel. • **global**—(Optional) Indicates a global pool tunnel. Entering this keyword is not necessary, for all tunnels are global pool in the absence of the sub-pool keyword. But if users of pre-DiffServ-aware Traffic Engineering (DS-TE) images enter this keyword, it is accepted. • *bandwidth*—Bandwidth, in kilobits per second, set aside for the MPLS traffic engineering tunnel. Range is between 1 and 4294967295. **Note** You can configure any existing **mpls traffic-eng** command on these TE or DS-TE tunnels.
Step 6	Router(config-if)# **tunnel mpls traffic-eng exp** [*list-of-exp-values*] [**default**]	Specifies an EXP value or values for an MPLS TE tunnel. • *list-of-exp-values*—EXP value or values that are are to be carried by the specified tunnel. Values range from 0 to 7. • **default**—The specified tunnel is to carry all EXP values that are: – Not explicitly allocated to another tunnel – Allocated to a tunnel that is currently down
Step 7	Router(config-if)# **exit**	Exits to global configuration mode.
Step 8		Repeat steps 1 through 7 on the same headend router to create additional tunnels from this headend to the same tailend.

Creating a Master Tunnel, Attaching Member Tunnels, and Making the Master Tunnel Visible

Perform the followings task to create a master tunnel, attach member tunnels to it, and make the master tunnel visible for routing. The procedure begins in global configuration mode.

	Command	Purpose
Step 1	Router(config)# **interface tunnel** *number*	Configures a tunnel interface type and enters interface configuration mode. • *number*—Number of the tunnel interface that you want to create or configure.
Step 2	Router(config-if)# **ip unnumbered** *type number*	Enables IP processing on an interface without assigning an explicit IP address to the interface. • *type*—Type of another interface on which the router has an assigned IP address. • *number*—Number of another interface on which the router has an assigned IP address. It cannot be another unnumbered interface.
Step 3	Router(config-if)# **tunnel destination** {*hostname* | *ip-address*}	Specifies the destination of the tunnel for this path option. • *hostname*—Name of the host destination. • *ip-address*—IP address of the host destination expressed in four-part, dotted decimal notation.
Step 4	Router(config-if)# **tunnel mode mpls traffic-eng exp-bundle master**	Specifies this is the master tunnel for the CBTS configuration.
Step 5	Router(config-if)# **tunnel mode mpls traffic-eng exp-bundle member** *tunnel-id*	Attaches a member tunnel to the master tunnel. • *tunnel-id*—Number of the tunnel interface to be attached to the master tunnel. Repeat this command for each member tunnel.

	Command	Purpose
Step 6	Router(config-if)# **tunnel mpls traffic-eng autoroute announce**	Specifies that the Interior Gateway Protocol (IGP) should use the tunnel (if the tunnel is up) in its enhanced SPF calculation.
Step 7	Router(config-if)# **tunnel mpls traffic-eng autoroute metric** {**absolute** | **relative**} *value*	(Optional) Specifies the MPLS TE tunnel metric that the IGP enhanced SPF calculation uses. • **absolute**—Indicates the absolute metric mode; you can enter a positive metric value. • **relative**—Indicates the relative metric mode; you can enter a positive, negative, or zero value. • *value*—Metric that the IGP enhanced SPF calculation uses. The relative value can be from –10 to 10. **Note** Even though the value for a relative metric can be from –10 to +10, configuring a tunnel metric with a negative value is considered a misconfiguration. If the metric to the tunnel tailend appears to be 4 from the routing table, then the cost to the tunnel tailend router is actually 3 because 1 is added to the cost for getting to the loopback address. In this instance, the lowest value that you can configure for the relative metric is -3.

Note Alternatively, static routing could be used instead of autoroute to make the TE or DS-TE tunnels visible for routing.

Verifying That the MPLS TE or DS-TE Tunnels Are Operating and Announced to the IGP

The following **show** commands can be used to verify that the MPLS TE or DS-TE tunnels are operating and announced to the IGP. The commands are all entered in privileged EXEC configuration mode.

Command	Purpose
Router# **show mpls traffic-eng topology** {*A.B*.C.*D* | **igp-id** {**isis** *nsap-address* | **ospf** *A.B*.C.*D*} [**brief**]	Shows the MPLS traffic engineering global topology as currently known at this node. • *A.B*.C.*D*—Specifies the node by the IP address (router identifier to interface address). • **igp-id**—Specifies the node by IGP router identifier. • **isis** *nsap-address*—Specifies the node by router identification (nsap-address) if you are using IS-IS. • **ospf** *A.B*.C.*D*—Specifies the node by router identifier if you are using OSPF. • **brief**—Provides a less detailed version of the topology.
Router# **show mpls traffic-eng exp**	Displays EXP mapping.

Command	Purpose
Router# **show ip cef** [*type number*] [**detail**]	Displays entries in the forwarding information base (FIB) or displays a summary of the FIB. • *type number* —Identifies the interface type and number for which to display FIB entries. • **detail**—Displays detailed FIB entry information.
Router# **show mpls forwarding-table** [*network* {*mask* | *length*} [**detail**]	Displays the contents of the MPLS label forwarding information base (LFIB). • *network*—Identifies the destination network number. • *mask*—Identifies the network mask to be used with the specified network. • *length*—Identifies the number of bits in the destination mask. • **detail**—Displays information in long form (includes length of encapsulation, length of MAC string, maximum transmission unit [MTU], and all labels).
Router# **show mpls traffic-eng autoroute**	Displays tunnels that are announced to the Interior Gateway Protocol (IGP).

The **show mpls traffic-eng topology** command output displays the MPLS TE global topology:

```
Router# show mpls traffic-eng topology 10.0.0.1

IGP Id: 10.0.0.1, MPLS TE Id:10.0.0.1 Router Node  (ospf 10  area 0) id 1
link[0]: Broadcast, DR: 180.0.1.2, nbr_node_id:6, gen:18
      frag_id 0, Intf Address:180.0.1.1
      TE metric:1, IGP metric:1, attribute_flags:0x0
      SRLGs: None
      physical_bw: 100000 (kbps), max_reservable_bw_global: 1000 (kbps)
      max_reservable_bw_sub: 0 (kbps)
                             Global Pool       Sub Pool
           Total Allocated   Reservable        Reservable
           BW (kbps)         BW (kbps)         BW (kbps)
               ---------------   -----------       ----------
        bw[0]:            0            1000                0
        bw[1]:            0            1000                0
        bw[2]:            0            1000                0
        bw[3]:            0            1000                0
        bw[4]:            0            1000                0
        bw[5]:            0            1000                0
        bw[6]:            0            1000                0
        bw[7]:          100             900                0

link[1]: Broadcast, DR: 180.0.2.2, nbr_node_id:7, gen:19
      frag_id 1, Intf Address:180.0.2.1
      TE metric:1, IGP metric:1, attribute_flags:0x0
      SRLGs: None
      physical_bw: 100000 (kbps), max_reservable_bw_global: 1000 (kbps)
          max_reservable_bw_sub: 0 (kbps)
                                 Global Pool       Sub Pool
               Total Allocated   Reservable        Reservable
               BW (kbps)         BW (kbps)         BW (kbps)
           ---------------   -----------       ----------
    bw[0]:            0            1000                0
    bw[1]:            0            1000                0
```

```
    bw[2]:            0             1000               0
    bw[3]:            0             1000               0
    bw[4]:            0             1000               0
    bw[5]:            0             1000               0
    bw[6]:            0             1000               0
    bw[7]:            0             1000               0
```

The **show mpls traffic-eng exp** command output displays EXP mapping information about a tunnel:

```
Router# show mpls traffic-eng exp

Destination: 10.0.0.9
    Master:Tunnel10Status: IP

    Members: StatusConf EXPActual EXP
    Tunnel1UP/ACTIVE55
    Tunnel2UP/ACTIVEdefault0 1 2 3 4 6 7
    Tunnel3UP/INACTIVE(T)2
    Tunnel4DOWN3
    Tunnel5UP/ACTIVE(NE)

(T)=Tailend is different to master
(NE)=There is no exp value configured on this tunnel.
```

The **show ip cef detail** command output displays detailed FIB entry information for a tunnel:

```
Router# show ip cef tunnel1 detail

IP CEF with switching (Table Version 46), flags=0x0
31 routes, 0 reresolve, 0 unresolved (0 old, 0 new), peak 2
2 instant recursive resolutions, 0 used background process
8 load sharing elements, 8 references
6 in-place/0 aborted modifications
34696 bytes allocated to the FIB table data structures
universal per-destination load sharing algorithm, id 9EDD49E1
1(0) CEF resets
    Resolution Timer: Exponential (currently 1s, peak 1s)
    Tree summary:
    8-8-8-8 stride pattern
    short mask protection disabled
31 leaves, 23 nodes using 26428 bytes
Table epoch: 0 (31 entries at this epoch)
Adjacency Table has 13 adjacencies
10.0.0.9/32, version 45, epoch 0, per-destination sharing
0 packets, 0 bytes
tag information set, all rewrites inherited
local tag: tunnel head
via 0.0.0.0, Tunnel1, 0 dependencies
traffic share 1
next hop 0.0.0.0, Tunnel1
valid adjacency
tag rewrite with Tu1, point2point, tags imposed {12304}
0 packets, 0 bytes switched through the prefix
tmstats: external 0 packets, 0 bytes
internal 0 packets, 0 bytes
```

The **show mpls forwarding-table detail** command output displays detailed information from the MPLS LFIB:

```
    Router# show mpls forwarding 10.0.0.9 detail

    Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
    tag    tag or VC   or Tunnel Id      switched   interface
Tun hd Untagged    10.0.0.9/32       0          Tu1        point2point
```

```
    MAC/Encaps=14/18, MRU=1500, Tag Stack{12304}, via Fa6/0
    00027D884000000ED70178A88847 03010000
    No output feature configured
Per-exp selection: 1
Untagged    10.0.0.9/32       0          Tu2        point2point
    MAC/Encaps=14/18, MRU=1500, Tag Stack{12305}, via Fa6/1
    00027D884001000ED70178A98847 03011000
    No output feature configured
Per-exp selection: 2  3
Untagged    10.0.0.9/32       0          Tu3        point2point
    MAC/Encaps=14/18, MRU=1500, Tag Stack{12306}, via Fa6/1
    00027D884001000ED70178A98847 03012000
    No output feature configured
Per-exp selection: 4  5
Untagged    10.0.0.9/32       0          Tu4        point2point
    MAC/Encaps=14/18, MRU=1500, Tag Stack{12307}, via Fa6/1
    00027D884001000ED70178A98847 03013000
    No output feature configured
Per-exp selection: 0  6  7
```

The **show mpls traffic-eng autoroute** command output displays tunnels that are announced to the Interior Gateway Protocol (IGP).

```
Router# show mpls traffic-eng autoroute

MPLS TE autorouting enabled
destination 10.0.0.9, area ospf 10  area 0, has 4 tunnels
Tunnel1     (load balancing metric 20000000, nexthop 10.0.0.9)
(flags: Announce)
Tunnel2     (load balancing metric 20000000, nexthop 10.0.0.9)
(flags: Announce)
Tunnel3     (load balancing metric 20000000, nexthop 10.0.0.9)
(flags: Announce)
Tunnel4     (load balancing metric 20000000, nexthop 10.0.0.9)
(flags: Announce)
```

Configuring QoS Features on a SIP

This section describes configuration of the SIP-specific QoS features using the Modular QoS command-line interface (CLI). Before referring to any other QoS documentation for the platform or in the Cisco IOS software, use this section to determine SIP-specific QoS feature support and configuration guidelines.

For additional details about QoS concepts and features in Cisco IOS 12.2 releases, you can then refer to the *Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2*, at

http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_book09186a00800c5e31.html

This section includes the following topics:

General QoS Feature Configuration Guidelines

This section identifies some general QoS feature guidelines for certain types of SPAs. You can find other feature-specific SIP and SPA configuration guidelines and restrictions in the other QoS sections of this chapter.

ATM SPA QoS Configuration Guidelines

For the 2-Port and 4-Port OC-3c/STM-1 ATM SPA, the following applies:

- In the ingress direction, all Quality of Service (QoS) features are supported by the Cisco 7600 SIP-200.
- In the egress direction:
  - All queueing-based features (such as class-based weighted fair queueing [CBWFQ], and ATM per-VC WFQ, WRED, and shaping) are implemented on the segmentation and reassembly (SAR) processor on the SPA.
  - Policing is implemented on the SIP.
  - Class queue shaping is not supported.

Ethernet SPA QoS Configuration Guidelines

For the Ethernet SPAs, the following QoS behavior applies:

- In both the ingress and egress directions, all QoS features calculate packet size similarly to how packet size calculation is performed by the FlexWAN and Enhanced FlexWAN modules on the Cisco 7600 series router.
- Specifically, all features consider the IEEE 802.3 Layer 2 headers and the Layer 3 protocol payload. The CRC, interframe gap, and preamble are not included in the packet size calculations.

Note For Fast Ethernet SPAs, QoS cannot change the speed of an interface (for example, Fast Ethernet SPAs cannot change QoS settings whenever an interface speed is changed between 100 and 10 Mbps). When the speed is changed, the user must also adjust the QoS setting accordingly.

Configuring QoS Features Using MQC

The Modular QoS CLI (MQC) is a CLI structure that allows users to create traffic policies and attach these policies to interfaces. A traffic policy contains a traffic class and one or more QoS features. A traffic class is used to select traffic, while the QoS features in the traffic policy determine how to treat the classified traffic.

If you apply a traffic policy at a main interface that also contains subinterfaces, then all of the traffic that goes through the subinterfaces is processed according to the policy at the main interface. For example, if you configure a traffic shaping policy at the main interface, all of the traffic going through the subinterfaces is aggregated and shaped to the rate defined in the traffic shaping policy at the main interface.

To configure QoS features using the Modular QoS CLI on the SIPs, complete the following basic steps:

Step 1 Define a traffic class using the **class-map** command.

Step 2 Create a traffic policy by associating the traffic class with one or more QoS features (using the **policy-map** command).

Step 3 Attach the traffic policy to the interface using the **service-policy** command.

For a complete discussion about MQC, refer to the "Modular Quality of Service Command-Line Interface Overview" chapter of the *Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2* publication at:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a00800bd908.html

Configuring QoS Traffic Classes on a SIP

Use the QoS classification features to select your network traffic and categorize it into classes for further QoS processing based on matching certain criteria. The default class, named "class-default," is the class to which traffic is directed for any traffic that does not match any of the selection criteria in the configured class maps.

QoS Traffic Class Configuration Guidelines

When configuring traffic classes on a SIP, consider the following guidelines:

- You can define up to 256 unique class maps.
- A single class map can contain up to 8 different **match** command statements.
- For ATM bridging, Frame Relay bridging, MPB, and BCP features, the following matching features are supported on bridged frames beginning in Cisco IOS Release 12.2(33)SRA:
  - Matching on ATM CLP bit (input interface only)
  - Matching on CoS
  - Matching on Frame Relay DE bit (input interface only)
  - Matching on Frame Relay DLCI
  - Matching on inner CoS
  - Matching on inner VLAN

- Matching on IP DSCP
- Matching on IP precedence
- Matching on VLAN

- The Cisco 7600 SIP-600 does not support combining matches on QoS group or input VLAN *with other types of matching criteria* (for example, access control lists [ACLs]) in the same class or policy map.
- The Cisco 7600 SIP-400 supports matching on ACLs for routed traffic only. Matching on ACLs is not supported for bridged traffic.
- The SIP-400 does not support dynamic, time-based, or tos-matching ACLs. The SIP-400 also does not support the log option in ACL.
- When configuring hierarchical QoS on the Cisco 7600 SIP-600, if you configure matching on an input VLAN in a parent policy, then only matching on a QoS group is supported in the child policy.
- For support of specific matching criteria by SIP, see Table 4-10.

To create a user-defined QoS traffic class, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **class-map** [**match-all** | **match-any**] *class-name*	Creates a traffic class, where: • **match-all**—(Optional) Specifies that all match criteria in the class map must be matched, using a logical AND of all matching statements defined under the class. This is the default. • **match-any**—(Optional) Specifies that one or more match criteria must match, using a logical OR of all matching statements defined under the class. • *class-name*—Specifies the user-defined name of the class. **Note** You can define up to 256 unique class maps.
Step 2	Router(config-cmap)# **match** *type*	Specifies the matching criterion to be applied to the traffic, where *type* represents one of the forms of the **match** command supported by the SIP as shown in Table 4-10. **Note** A single class-map can contain up to 8 different **match** command statements.

Table 4-10 provides information about which QoS classification features are supported for SIPs on the Cisco 7600 series router. For more information about most of the commands documented in this table, refer to the *Cisco IOS Quality of Service Solutions Command Reference.*

Table 4-10 QoS Classification Feature Compatibility by SIP

Feature (match command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Matching on access control list (ACL) number (**match access-group** command)	Supported for all SPAs with the following types of ACLs: • Protocols—ICMP, IGMP, EIGRP, OSPF, PIM, and GRE • Source and destination port • TCP flags • ToS (DSCP and precedence)	Supported for all SPAs with the following types of ACLs: • Source and destination port • TCP flag (IPv4 only) • IP address (IPv6 compress mode only)	Supported for all SPAs with the following types of ACLs: • IPv4 and IPv6 • Protocols—ICMP, IGMP, UDP, and MAC • Source and destination ports • TCP flags • ToS
Matching on ACL name (**match access-group name** command)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs.
Match on any packet (**match any** command) **Note** Not supported for user-defined class maps.	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs.
Matching on ATM cell loss priority (CLP) (**match atm clp** command)	• Supported for all ATM SPAs. • Cisco IOS Release 12.2(33)SRA—Support added for ATM CLP matching with RFC 1483 bridging features.	• Supported for all ATM SPAs on ATM input interface only. • Cisco IOS Release 12.2(33)SRA—Support added for ATM CLP matching with RFC 1483 bridging features on ATM input interface only.	Not supported.
Matching on class map (**match class-map** command)	Supported for all SPAs.	Not supported.	Not supported.
Matching on Class of Service (CoS) (**match cos** command)	Supported in Cisco IOS Release 12.2(33)SRA on the 4-Port and 8-Port Fast Ethernet SPA using dot1q encapsulation.	• 2-Port Gigabit Ethernet SPA only—Input and output 802.1Q tagged frames. • Cisco IOS Release 12.2(33)SRA—Support added for inner CoS matching with bridging features.	Supported in Cisco IOS Release 12.2(33)SRA for switchport queueing. **Note** CoS classification is available through PFC QoS using MAC address ACLs.

Table 4-10 QoS Classification Feature Compatibility by SIP (continued)

Feature (match command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Matching on inner CoS (**match cos inner** command)	• Supported for all SPAs. • Cisco IOS Release 12.2(33)SRA—Supported added for inner CoS matching with bridging features.	Supported in Cisco IOS Release 12.2(33)SRA on the 2-Port Gigabit Ethernet SPA: • Input and output interfaces • Inner CoS matching with bridging features	Not supported.
Match on Frame Relay discard eligibility (DE) bit (**match fr-de** command)	• Supported for Frame Relay input and output interfaces. • Cisco IOS Release 12.2(33)SRA—Support added for Frame Relay DE matching with Frame Relay bridging features.	• Supported for a Frame Relay input interface only. • Cisco IOS Release 12.2(33)SRA—Support added for Frame Relay DE matching with Frame Relay bridging features on input Frame Relay interface only. **Note** Because the Cisco 7600 SIP-400 acts as a Frame Relay data terminal equipment (DTE) device only, and not a data communications equipment (DCE) device, the Cisco 7600 SIP-400 does not support dropping of frames that match on FR DE bits; however, other QoS actions are supported.	Not supported.
Match on Frame Relay data-link connection identifier (DLCI) (**match fr-dlci** command)	• Supported for Frame Relay input and output interfaces. • Cisco IOS Release 12.2(33)SRA—Support added for Frame Relay DLCI matching with Frame Relay bridging features.	Supported in Cisco IOS Release 12.2(33)SRA on Frame Relay input and output interfaces, and with Frame Relay bridging features.	Not supported.

Table 4-10 QoS Classification Feature Compatibility by SIP (continued)

Feature (match command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Match on input VLAN (**match input vlan** command—Matches the VLAN from an input interface)	Supported for EoMPLS interfaces.	Supported in Cisco IOS Release 12.2(33)SRA—Output interface only, and with bridging features. **Note** Service policy is applied on the output interface of the Cisco 7600 SIP-400 to match the VLAN from the input interface.	Supported in Cisco IOS Release 12.2(33)SRA—Output interface only for software-based EoMPLS. **Note** The service policy is applied on the output interface of the Cisco 7600 SIP-600 to match the VLAN from the input interface. If you configure matching on an input VLAN in a parent policy with hierarchical QoS, then only matching on QoS group is supported in the child policy.
Match on IP DSCP (**match ip dscp** command)	• Supported for all SPAs. • Cisco IOS Release 12.2(33)SRA—Support added for IP DSCP matching with bridging features on an input interface only.	• Supported for all SPAs. • Cisco IOS Release 12.2(33)SRA—Support added for IP DSCP matching with bridging features.	Supported for all SPAs.
Match on IP precedence (**match ip precedence** command)	Supported for all SPAs.	• Supported for all SPAs. • Cisco IOS Release 12.2(33)SRA—Support added for IP precedence matching with bridging features.	Supported for all SPAs.
Match on IP Real-Time Protocol (RTP) (**match ip rtp** command)	Supported for all SPAs.	Not supported.	Not supported.
Match on MAC address for an ACL name (**match mac address** command)	Not supported.	Not supported.	Not supported.
Match on destination MAC address (**match destination-address mac** command)	Not supported.	Not supported.	Not supported.

Table 4-10 QoS Classification Feature Compatibility by SIP (continued)

Feature (match command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Match on source MAC address (**match source-address mac** command)	Not supported.	Not supported.	Not supported.
Match on MPLS experimental (EXP) bit (**match mpls experimental** command)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs.
Match on Layer 3 packet length in IP header (**match packet length** command)	Supported for all SPAs.	Not supported.	Not supported.
Match on QoS group (**match qos-group** command)	Not supported.	Supported in Cisco IOS Release 12.2(33)SRA—Output interface only.	Supported in software-based EoMPLS configurations only using hierarchical QoS, where the parent policy configures matching on input VLAN and the child policy configures matching on QoS group.
Match on protocol (**match protocol** command	Supported for NBAR.	Not supported.	Supports matching on IP and IPv6.
Match on VLAN (**match vlan** command—Matches the outer VLAN of a Layer 2 802.1Q frame)	Not supported.	Supported in Cisco IOS Release 12.2(33)SRA: • Input and output interfaces • Outer VLAN ID matching for 802.1Q tagged frames	Supported in Cisco IOS Release 12.2(33)SRA: • Output interface only • Outer VLAN ID matching for 802.1Q tagged frames
Match on VLAN Inner (**match vlan inner** command—Matches the innermost VLAN of the 802.1Q tag in the Layer 2 frame)	• Supported for all SPAs. • Cisco IOS Release 12.2(33)SRA—Support added for inner VLAN ID matching with bridging features.	Supported in Cisco IOS Release 12.2(33)SRA: • Input and output interface • Inner VLAN ID matching with bridging features	Not supported.
No match on specified criteria (**match not** command)	Supported for all SPAs.	Supported for all SPAs.	Not supported.

Configuring QoS Class-Based Marking Policies on a SIP

After you have created your traffic classes, you can configure traffic policies to configure marking features to apply certain actions to the selected traffic in those classes.

In most cases, the purpose of a packet mark is identification. After a packet is marked, downstream devices identify traffic based on the marking and categorize the traffic according to network needs. This categorization occurs when the **match** commands in the traffic class are configured to identify the packets by the mark (for example, **match ip precedence**, **match ip dscp**, **match cos**, and so on). The traffic policy using this traffic class can then set the appropriate QoS features for the marked traffic.

In some cases, the markings can be used for purposes besides identification. Distributed WRED, for instance, can use the IP precedence, IP DSCP, or MPLS EXP values to detect and drop packets. In ATM networks, the CLP bit of the packet is used to determine the precedence of packets in a congested environment. If congestion occurs in the ATM network, packets with the CLP bit set to 1 are dropped before packets with the CLP bit set to 0. Similarly, the DE bit of a Frame Relay frame is used to determine the priority of a frame in a congested Frame Relay network. In Frame Relay networks, frames with the DE bit set to 1 are dropped before frames with the DE bit set to 0.

QoS Class-Based Marking Policy Configuration Guidelines

When configuring class-based marking on a SIP, consider the following guidelines:

- Packet marking is supported on interfaces, subinterfaces, and ATM virtual circuits (VCs). In an ATM PVC, you can configure packet marking in the same traffic policy where you configure the queueing actions, on a per-VC basis. However, only PVC configuration of service policies is supported for classes using multipoint bridging (MPB) match criteria.
- For ATM bridging, Frame Relay bridging, MPB, and BCP features, the following marking features are supported on bridged frames beginning in Cisco IOS Release 12.2(33)SRA:
  - Set ATM CLP bit (output interface only)
  - Set Frame Relay DE bit (output interface only)
  - Set inner CoS
- If a service policy configures both class-based marking and marking as part of a policing action, then the marking using policing takes precedence over any class-based marking.
- The Cisco 7600 SIP-600 supports marking on input interfaces only.
- For support of specific marking criteria by SIP, see Table 4-11.

To configure a QoS traffic policy with class-based marking, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **policy-map** *policy-map-name*	Creates or modifies a traffic policy and enters policy map configuration mode, where: • *policy-map-name*—Specifies the name of the traffic policy to configure. Names can be a maximum of 40 alphanumeric characters.

	Command	Purpose
Step 2	Router (config-pmap)# **class** {*class-name* | **class-default**}	Specifies the name of the traffic class to which this policy applies and enters policy-map class configuration mode, where: • *class-name*—Specifies that the policy applies to a user-defined class name previously configured. • **class-default**—Specifies that the policy applies to the default traffic class.
Step 3	Router(config-pmap-c)# **set** *type*	Specifies the marking action to be applied to the traffic, where *type* represents one of the forms of the **set** command supported by the SIP as shown in Table 4-11.

Table 4-11 provides information about which QoS class-based marking features are supported for SIPs on the Cisco 7600 series router.

Table 4-11 QoS Class-Based Marking Feature Compatibility by SIP

Marking Feature (set command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Set ATM CLP bit (**set atm-clp** command—Marks the ATM cell loss bit with value of 1)	• Supported for ATM output interfaces only. • Cisco IOS Release 12.2(33)SRA—Support added for ATM CLP marking on output interfaces only with RFC 1483 bridging features.	Supported for ATM SPA output interfaces only.	Not supported.
Set discard class (**set discard-class** command—Marks the packet with a discard class value for per-hop behavior)	Not supported.	Not supported.	Not supported.
Set Frame Relay DE bit (**set fr-de** command—Marks the Frame Relay discard eligibility bit with value of 1)	• Supported for Frame Relay output interfaces only. • Cisco IOS Release 12.2(33)SRA—Support added for Frame Relay DE marking on output interfaces only with Frame Relay bridging features.	Supported for Frame Relay output interfaces only.	Not supported.
Set IP DSCP (**set ip dscp** command—Marks the IP differentiated services code point [DSCP] in the type of service [ToS] byte with a value from 0 to 63)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs on an input interface.

Table 4-11 QoS Class-Based Marking Feature Compatibility by SIP (continued)

Marking Feature (set command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Set IP precedence (**set ip precedence** command—Marks the precedence value in the IP header with a value from 0 to 7.)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs on an input interface.
Set Layer 2 802.1Q CoS (**set cos** command—Marks the CoS value from 0 to 7 in an 802.1Q tagged frame)	• Supported for all SPAs. • In Cisco IOS Release 12.2(33)SRA—Not supported with **set cos-inner** command on the same interface.	Supported in Cisco IOS Release 12.2(33)SRA.	Not supported.
Set Layer 2 802.1Q CoS (**set cos-inner** command—Marks the inner CoS field from 0 to 7 in a bridged frame)	Supported in Cisco IOS Release 12.2(33)SRA with bridging features on the 4-Port and 8-Port Fast Ethernet SPA.	Supported in Cisco IOS Release 12.2(33)SRA with bridging features.	Not supported.
Set MPLS experimental (EXP) bit on label imposition (**set mpls experimental imposition** command)	Supported for all SPAs.	Supported for any SPA IP input interface. **Note** The **table** keyword is not supported.	Supported for all SPAs on an input interface.
Set MPLS EXP topmost (**set mpls experimental topmost** command)	Supported for all SPAs.	Supported for any SPA MPLS interface.	Not supported.
Set QoS group (**set qos-group** command—Marks the packet with a QoS group association)	Not supported.	Not supported.	Supported only for software-based EoMPLS on an input SPA switchport interface.

For more detailed information about configuring class-based marking features, refer to the *Class-Based Marking* document located at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t5/cbpmark2.htm

Note When referring to other class-based marking documentation, be sure to note any SIP-specific configuration guidelines described in this document.

Configuring QoS Congestion Management and Avoidance Policies on a SIP

This section describes SIP- and SPA-specific information for configuring QoS traffic policies for congestion management and avoidance features. These features are generally referred to as queueing features.

QoS Congestion Management and Avoidance Policy Configuration Guidelines

When configuring queueing features on a SIP, consider the following guidelines:

- The Cisco 7600 series router supports different forms of queueing features. See Table 4-12 to determine which queueing features are supported by SIP type.
- The Cisco 7600 SIP-200 and Cisco 7600 SIP-400 do not support ingress queueing features.
- When configuring queueing on the Cisco 7600 SIP-400, consider the following guidelines:
  - A queue on the Cisco 7600 SIP-400 is not assured any minimum bandwidth.
  - You cannot configure bandwidth or shaping with queueing under the same class in a service policy on the Cisco 7600 SIP-400.
  - If you want to define bandwidth parameters under different classes in the same service policy on the Cisco 7600 SIP-400, then you only can use the **bandwidth remaining percent** command. The Cisco 7600 SIP-400 does not support other forms of the **bandwidth** command with queueing in the same service policy.
- You can use policing with queueing to limit the traffic rate.
- On the Cisco 7600 SIP-400, WRED is supported on bridged VCs with classification on precedence and DSCP values. On other SIPs, WRED does not work on bridged VCs (for example, VCs that implement MPB).
- When configuring WRED on the Cisco 7600 SIP-400, consider the following guidelines:
  - WRED is supported on bridged VCs with classification on precedence and DSCP values.
  - WRED explicit congestion notification (ECN) is not supported for output traffic on ATM SPAs.
  - ECN is supported for IP traffic on output POS interfaces only.
  - You can use the low-order TOS bits in the IP header for explicit congestion notification (ECN) for WRED. If you configure **random-detect ecn** in a service policy and apply it to either a POS interface or a VC on a POS interface, then if at least one of the ECN bits is set and the packet is a candidate for dropping, the Cisco 7600 SIP-400 marks both ECN bits. If either one of the ECN bits is set, the Cisco 7600 SIP-400 will not drop the packet.
  - WRED ECN is not support for MPLS packets.
- On the Cisco 7600 SIP-400, the default queue limit is calculated based on the number of 250-byte packets that the SIP can transmit in one half of a second. For example, for an OC-3 SPA with a rate of 155 Mbps, the default queue limit is 38,750 packets (155000000 x 0.5 / 250 x 8).
- For more detailed information about configuring congestion management features, refer to the *Cisco IOS Quality of Service Solutions Configuration Guide* document corresponding to your Cisco IOS software release.

Table 4-12 provides information about which QoS queueing features are supported for SIPs on the Cisco 7600 series router.

Table 4-12 QoS Congestion Management and Avoidance Feature Compatibility by SIP and SPA Combination

Congestion Management and Avoidance Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Aggregate Weighted Random Early Detection (**random-detect aggregate**, **random-detect dscp (aggregate)**, and **random-detect precedence (aggregate)** commands)	Supported for ATM SPA PVCs only—Cisco IOS Release 12.2(18)SXE and later and in Cisco IOS Release 12.2(33)SRA	Supported for ATM SPA PVCs only—Cisco IOS Release 12.2(18)SXE and later and in Cisco IOS Release 12.2(33)SRA.	Supported for all SPAs. For more information on configuring aggregate WRED, see the "Configuring Aggregate WRED for PVCs" section on page 7-27.
Class-based Weighted Fair Queueing (CBWFQ) (**bandwidth**, **queue-limit** commands)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs.
Dual-Queue Support (**priority** and **priority level** commands)	Not supported.	Supported for all SPAs—Cisco IOS Release 12.2(33)SRB and later.	Not supported.
Flow-based Queueing (fair queueing/WFQ) (**fair-queue** command)	Supported for all SPAs.	Not supported.	Not supported.
Low Latency Queueing (LLQ)/ Queueing (**bandwidth** command)	Strict priority only—Supported for all SPAs.	Strict priority only—Supported for all SPAs.	Supported for all SPAs.
Random Early Detection (RED) (**random-detect** commands)	Supported for all SPAs.	Supported for all SPAs. • ATM SPAs—Up to 106 unique WRED minimum threshold (min-th), maximum threshold (max-th), and mark probability profiles supported. • Other SPAs—Up to 128 unique WRED min-th, max-th, and mark probability profiles supported.	Not supported.
Weighted RED (WRED)	Supported for all SPAs, with the following exception: • WRED is not supported on bridged VCs.	Supported for all SPAs, with the following restriction: • WRED is supported on bridged VCs with classification on precedence and DSCP values.	Not supported.

To configure a QoS CBWFQ policy, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **policy-map** *policy-map-name*	Creates or modifies a traffic policy and enters policy map configuration mode, where: • *policy-map-name*—Specifies the name of the traffic policy to configure. Names can be a maximum of 40 alphanumeric characters.
Step 2	Router (config-pmap)# **class** {*class-name* | **class-default**}	Specifies the name of the traffic class to which this policy applies and enters policy-map class configuration mode, where: • *class-name*—Specifies that the policy applies to a user-defined class name previously configured. • **class-default**—Specifies that the policy applies to the default traffic class.
Step 3	Router(config-pmap-c)# **bandwidth** {*bandwidth-kbps* | **percent** *percent*}	Specifies the bandwidth allocated to a class belonging to a policy map. **Note** The amount of bandwidth configured should be large enough to also accommodate Layer 2 overhead. • *bandwidth-kbps*—Specifies the amount of bandwidth, in number of kbps, to be assigned to a class. • **percent**—Specifies the amount of guaranteed bandwidth, based on the absolute percent of available bandwidth. • *percentage*—Used in conjunction with the percent keyword, the percentage of the total available bandwidth to be set aside for the priority classes.
Step 4	Router(config-pmap-c)# **queue-limit** *number-of-packets*	Specifies the maximum number of packets the queue can hold for a class policy configured in a policy map. • *number-of-packets*—A number in the range 1 to 64 specifying the maximum number of packets that the queue for this class can accumulate.

Configuring Dual Priority Queuing on a Cisco 7600 SIP-400

When configuring Dual Priority Queuing, consider the following guidelines:

- Only two priority levels are supported.
- Level 1 is higher than level 2.
- Propagation is supported on both levels.
- A priority without a level is mapped to level 1.

- The sum of bandwidth percentage and another queues' bandwidth reservation must not exceed 100% bandwidth.
- The police rate includes a Layer 2 header but not cyclic redundancy check (CRC), preamble, or interframe gap.
- Dual priority queuing is not supported on ATM SPAs.

To configure dual priority queuing, use the following commands:

Command or Action	Purpose
Router(config-pmap-c)# **priority**	Gives priority to a class of traffic belonging to a policy map.
Router(config-pmap-c)# **priority** *level*	Configures multiple priority queues. • *level*—A range of priority levels. Valid values are from 1 (high priority) to 4 (low priority). The default is 1.
Router(config-pmap-c)# **priority** *y ms*	• *ms*—Specifies the burst size in bytes. The burst size configures the network to accommodate temporary bursts of traffic.
Router(config-pmap-c)# **priority** *x kbps y bytes*	• *x kbps*—Specifies the burst size in kbps. • *y bytes*—Specifies the burst size in bytes.
Router(config-pmap-c)# **priority percent** *x%* | *y ms*	Enables conditional policing rate (kbps or link percent). Conditional policing is used if the logical or physical link is congested.

Configuring Percent Priority and Percent Bandwidth Support on a Cisco 7600 SIP-400

To configure percent priority and percent bandwidth, use the following commands:

Command or Action	Purpose
Router(config-pmap-c)# **bandwidth** *x kbps*	Specifies or modifies the bandwidth allocated for a class belonging to a policy map.
Router(config-pmap-c)# **bandwidth** percent *x%*	Specifies the amount of guaranteed bandwidth, based on an absolute percent of available bandwidth.
Router(config-pmap-c)# **bandwidth remaining percent** *x%*	• **remaining percent**—Amount of guaranteed bandwidth, based on a relative percent of available bandwidth.

Configuring QoS Traffic Shaping Policies on a SIP

This section describes SIP- and SPA-specific information for configuring QoS traffic policies for shaping traffic.

QoS Traffic Shaping Policy Configuration Guidelines

When configuring queueing features on a SIP, consider the following guidelines:

- The Cisco 7600 series router supports different forms of queueing features. See Table 4-13 to determine which traffic shaping features are supported by SIP type.
- Use a hierarchical policy if you want to achieve minimum bandwidth guarantees using CBWFQ with a Frame Relay map class. First, configure a parent policy to shape to the total bandwidth required (on the Cisco 7600 SIP-400, use the class-default in Cisco IOS Release 12.2(18)SXF, or a user-defined class beginning in Cisco IOS Release 12.2(33)SRA). Then, define a child policy using CBWFQ for the minimum bandwidth percentages.
- ATM SPAs do not support MQC-based traffic shaping. You need to configure traffic shaping for ATM interfaces using ATM Layer 2 VC shaping.
- For more detailed information about configuring congestion management features, refer to the *Cisco IOS Quality of Service Solutions Configuration Guide* document corresponding to your Cisco IOS software release.

Table 4-13 provides information about which QoS traffic shaping features are supported for SIPs on the Cisco 7600 series router.

Table 4-13 QoS Traffic Shaping Feature Compatibility by SIP and SPA Combination

Traffic Shaping Feature (shape command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Adaptive shaping for Frame Relay (**shape adaptive** command)	Supported for all SPAs.	Not supported.	Not supported.
Class-based shaping (**shape average**, **shape peak** commands)	Supported for all SPAs.	Supported for all SPAs, with the following exceptions: • Committed burst (bc)—Not supported. • Excess burst (be)—Not supported.	Supports **shape average** only for all SPAs.
Policy-map class shaping of average-rate of traffic by percentage of bandwidth (**shape average percent command**)	Not supported.	Supported for all SPAs.	Not supported.
Policy-map class shaping with adaptation to backward explicit congestion notification (BECN) (**shape adaptive** command)	Supported for all SPAs.	Not supported.	Not supported.

Table 4-13 QoS Traffic Shaping Feature Compatibility by SIP and SPA Combination (continued)

Traffic Shaping Feature (shape command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Policy-map class shaping with reflection of forward explicit congestion notification (FECN) as BECN (**shape fecn-adapt** command)	Supported for all SPAs.	Not supported.	Not supported.
Policy-map class shaping of peak-rate of traffic by percentage of bandwidth (**shape peak percent** command)	Not supported.	Not supported.	Not supported.

Configuring QoS Traffic Policing Policies on a SIP

This section describes SIP- and SPA-specific information for configuring QoS traffic policing policies.

QoS Traffic Policing Policy Configuration Guidelines

When configuring traffic policing on a SIP, consider the following guidelines:

- The Cisco 7600 series router supports different forms of policing using the **police** command. See Table 4-14 to determine which policing features are supported by SIP type.
- When configuring policing on the Cisco 7600 SIP-600, consider the following guidelines:
  - The Cisco 7600 SIP-600 supports **conform-action** policing on input interfaces only, unless it is being implemented with queueing.
  - The Cisco 7600 SIP-600 does not support any policing actions (shown in Table 4-15) using the **exceed-action** or **violate-action** keywords on an input interface.
  - The Cisco 7600 SIP-600 supports **exceed-action** policing on an output interface with a **drop** action only, when the policing is being implemented with queueing.
  - The Cisco 7600 SIP-600 supports marking for **exceed-action** policing only using the **set-dscp-transmit** command.
- When configuring a policing service policy and specifying the CIR in bits per second without specifying the optional conform (bc) or peak (be) burst in bytes, the Cisco 7600 SIP-400 calculates the burst size based on the number of bytes that it can transmit in 250 ms using the CIR value. For example, a CIR of 1 Mbps (or 1,000,000 bps) is equivalent to 125,000 bytes per second, which is 125 bytes per millisecond. The calculated burst is 250 x 125 = 31250 bytes. If the calculated burst is less than the interface maximum transmission unit (MTU), then the interface MTU is used as the burst size.
- You can use policing with queueing to limit the traffic rate.
- If a service policy configures both class-based marking and marking as part of a policing action, then the marking using policing takes precedence over any class-based marking.
- When configuring policing with MPB features on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, the **set-cos-inner-transmit** action is supported beginning in Cisco IOS Release 12.2(33)SRA.

Table 4-14 provides information about which policing features are supported for SIPs on the Cisco 7600 series router.

Table 4-14 QoS Policing Feature Compatibility by SIP and SPA Combination

Policing Feature (police command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Policing by aggregate policer (**police aggregate** command)	Not supported.	Not supported.	Supported for all SPAs.
Policing by bandwidth using token bucket algorithm (**police** command)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAS.
Policing by committed information rate (CIR) percentage (**police (percent)** command—**police cir percent** form)	Supported for all SPAs.	Supported for all SPAs.	Not supported.
Policing with 2-color marker (CIR and peak information rate [PIR]) (**police (two rates)** command—**police cir pir** form)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs.
Policing by flow mask (**police flow mask** command)	Not supported.	Not supported.	Supported for all SPAs.
Policing by microflow (**police flow** command)	Not supported.	Not supported.	Supported for all SPAs.

To create QoS traffic policies with policing, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **policy-map** *policy-map-name*	Creates or modifies a traffic policy and enters policy map configuration mode, where: • *policy-map-name*—Specifies the name of the traffic policy to configure. Names can be a maximum of 40 alphanumeric characters.
Step 2	Router (config-pmap)# **class** {*class-name* | **class-default**}	Specifies the name of the traffic class to which this policy applies and enters policy-map class configuration mode, where: • *class-name*—Specifies that the policy applies to a user-defined class name previously configured. • **class-default**—Specifies that the policy applies to the default traffic class.
Use one of the following forms of **police** commands to evaluate traffic for the specified class. See Table 4-14 to determine which SIPs support the different policing features.		

	Command	Purpose
Step 3	Router(config-pmap-c)# **police** *bps* [*burst-normal*] [*burst-max*] **conform-action** *action* **exceed-action** *action* **violate-action** *action*	Specifies a maximum bandwidth usage by a traffic class through the use of a token bucket algorithm, where: • *bps*—Specifies the average rate in bits per second. Valid values are 8000 to 200000000. • *burst-normal*—(Optional) Specifies the normal burst size in bytes. Valid values are 1000 to 51200000. The default normal burst size is 1500 bytes. • *burst-max*—(Optional) Specifies the excess burst size in bytes. Valid values are 1000 to 51200000. • *action*—Specifies the policing command (as shown in Table 4-15) for the action to be applied to the corresponding conforming, exceeding, or violating traffic.
Step 4	Router(config-pmap-c)# **police cir percent** *percentage* [*burst-in-msec*] [**bc** *conform-burst-in-msec*] [**pir percent** *percentage*] [**be** *peak-burst-in-msec*] [**conform-action** *action* [**exceed-action** *action* [**violate-action** *action*]]]	Configures traffic policing on the basis of a percentage of bandwidth available on an interface, where: • **cir percent** *percentage*—Specifies the committed information rate (CIR) bandwidth percentage. Valid values are 1 to 100. • *burst-in-msec*—(Optional) Burst in milliseconds. Valid values are 1 to 2000. • **bc** *conform-burst-in-msec*—(Optional) Specifies the conform burst (bc) size used by the first token bucket for policing traffic in milliseconds. Valid values are 1 to 2000. • **pir percent** *percentage*—(Optional) Specifies the peak information rate (PIR) bandwidth percentage. Valid values are 1 to 100. • **be** *peak-burst-in-msec*—(Optional) Specifies the peak burst (be) size used by the second token bucket for policing traffic in milliseconds. Valid values are 1 to 2000. • *action*—Specifies the policing command (as shown in Table 4-15) for the action to be applied to the corresponding conforming, exceeding, or violating traffic.

	Command	Purpose
Step 5	Router(config-pmap-c)# **police** {**cir** *cir*} [**bc** *conform-burst*] {**pir** *pir*} [**be** *peak-burst*] [**conform-action** *action* [**exceed-action** *action* [**violate-action** *action*]]]	Configures traffic policing using two rates, the committed information rate (CIR) and the peak information rate (PIR), where: • **cir** *cir*—Specifies the CIR at which the first token bucket is updated as a value in bits per second. Valid values are 8000 to 200000000. • **bc** *conform-burst*—(Optional) Specifies the conform burst (bc) size in bytes used by the first token bucket for policing. Valid values are 1000 to 51200000. • **pir** *pir*—Specifies the PIR at which the second token bucket is updated as a value in bits per second. Valid values are 8000 to 200000000. • **be** *peak-burst*—(Optional) Specifies the peak burst (be) size in bytes used by the second token bucket for policing. The size varies according to the interface and platform in use. • *action*—(Optional) Specifies the policing command (as shown in Table 4-15) for the action to be applied to the corresponding conforming, exceeding, or violating traffic.
Step 6	Router(config-pmap-c)# **police flow** {*bits-per-second* [*normal-burst-bytes*] [*maximum-burst-bytes*] [**pir** *peak-rate-bps*]} | [**conform-action** *action*] [**exceed-action** *action*] [**violate-action** *action*]	Configures a microflow policer, where: • *bits-per-second*—Specifies the CIR in bits per second. Valid values are from 32000 to 4000000000 bits per second. • *normal-burst-bytes*—(Optional) Specifies the CIR token bucket size. Valid values are from 1000 to 512000000 bytes. • *maximum-burst-bytes*—(Optional) Specifies the PIR token-bucket size. Valid values are from 1000 to 32000000 bytes. • **pir** *peak-rate-bps*—(Optional) Specifies the PIR in bits per second. Valid values are from 32000 to 4000000000 bits per second. • *action*—Specifies the policing command (as shown in Table 4-15) for the action to be applied to the corresponding conforming, exceeding, or violating traffic.

	Command	Purpose
Step 7	Router(config-pmap-c)# **police flow mask** {**dest-only** | **full-flow** | **src-only**} {*bits-per-second* [*normal-burst-bytes*] [*maximum-burst-bytes*]} [**conform-action** *action*] [**exceed-action** *action*]	Configures a flow mask to be used for policing, where: • **dest-only**—Specifies the destination-only flow mask. • **full-flow**—Specifies the full-flow mask. • **src-only**—Specifies the source-only flow mask. • *bits-per-second*—Specifies the CIR in bits per second. Valid values are from 32000 to 4000000000 bits per second. • *normal-burst-bytes*—(Optional) Specifies the CIR token bucket size. Valid values are from 1000 to 512000000 bytes. • *maximum-burst-bytes*—(Optional) Specifies the PIR token bucket size. Valid values are from 1000 to 32000000 bytes. • *action*—Specifies the policing command (as shown in Table 4-15) for the action to be applied to the corresponding conforming or exceeding traffic.
Step 8	Router(config-pmap-c)# **police aggregate** *name*	Specifies a previously defined aggregate policer name and configures the policy-map class to use the specified *name* of the aggregate policer.

Table 4-15 provides information about which policing actions are supported for SIPs on the Cisco 7600 series router.

Note For restrictions on use of certain marking features with different types of policing actions (conform, exceed, or violate actions), be sure to see the "QoS Traffic Policing Policy Configuration Guidelines" section on page 4-77.

Table 4-15 QoS Policing Action Compatibility by SIP and SPA Combination

Policing Action (set command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Drop the packet (**drop** command)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs—Input interface only.
Set the ATM CLP bit to 1 and transmit (**set-clp-transmit** command)	Supported for all SPAs.	Supported for all SPAs.	Not supported.
Set the inner CoS value and transmit (**set-cos-inner-transmit** command)	Supported in Cisco IOS Release 12.2(33)SRA with bridging features.	Supported in Cisco IOS Release 12.2(33)SRA with bridging features.	Not supported.
Set the Frame Relay DE bit to 1 and transmit (**set-frde-transmit** command)	Supported for all SPAs.	Supported for all SPAs.	Not supported.

Table 4-15 QoS Policing Action Compatibility by SIP and SPA Combination (continued)

Policing Action (set command)	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Set the IP precedence and transmit (**set-prec-transmit** command)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs—Input interface only.
Set the IP DSCP and transmit (**set-dscp-transmit** command)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs—Input interface only.
Set the MPLS EXP bit (0–7) on imposition and transmit (**set-mpls-experimental-imposition-transmit** command	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs.
Set the MPLS EXP bit in the topmost label and transmit (**set-mpls-experimental-topmost-transmit** command)	Supported for all SPAs.	Supported for all SPAs.	Supported for all SPAs.
Transmit all packets without alteration (**transmit** command)	Supported for all SPAs.	Supported for all SPAs	Supported for all SPAs.

Attaching a QoS Traffic Policy to an Interface

Before a traffic policy can be enabled for a class of traffic, it must be configured on an interface. A traffic policy also can be attached to an ATM permanent virtual circuit (PVC) subinterface, Frame Relay data-link connection identifier (DLCI), and Ethernet subinterfaces.

Traffic policies can be applied for traffic coming into an interface (input), and for traffic leaving that interface (output).

Attaching a QoS Traffic Policy for an Input Interface

When you attach a traffic policy to an input interface, the policy is applied to traffic coming into that interface. To attach a traffic policy for an input interface, use the following command beginning in interface configuration mode:

Command	Purpose
Router(config-if)# **service-policy input** *policy-map-name*	Attaches a traffic policy to the input direction of an interface, where: • *policy-map-name*—Specifies the name of the traffic policy to configure.

Attaching a QoS Traffic Policy to an Output Interface

When you attach a traffic policy to an output interface, the policy is applied to traffic leaving that interface. To attach a traffic policy to an output interface, use the following command beginning in interface configuration mode:

Command	Purpose
Router(config-if)# **service-policy output** *policy-map-name*	Attaches a traffic policy to the output direction of an interface, where: • *policy-map-name*—Specifies the name of the traffic policy to configure.

Configuring Network-Based Application Recognition and Distributed Network-Based Application Recognition

Note Network-Based Application Recognition (NBAR) and Distributed Network-Based Application Recognition (dNBAR) are supported on the Cisco 7600 SIP-200 only.

The purpose of IP quality of service (QoS) is to provide appropriate network resources (bandwidth, delay, jitter, and packet loss) to applications. QoS maximizes the return on investments on network infrastructure by ensuring that mission-critical applications get the required performance and noncritical applications do not hamper the performance of critical applications.

IP QoS can be deployed by defining classes or categories of applications. These classes are defined by using various classification techniques available in Cisco IOS software. After these classes are defined and attached to an interface, the desired QoS features, such as marking, congestion management, congestion avoidance, link efficiency mechanisms, or policing and shaping can then be applied to the classified traffic to provide the appropriate network resources amongst the defined classes.

Classification, therefore, is an important first step in configuring QoS in a network infrastructure.

NBAR is a classification engine that recognizes a wide variety of applications, including web-based and other difficult-to-classify protocols that utilize dynamic TCP/UDP port assignments. When an application is recognized and classified by NBAR, a network can invoke services for that specific application. NBAR ensures that network bandwidth is used efficiently by classifying packets and then applying QoS to the classified traffic. Some examples of class-based QoS features that can be used on traffic after the traffic is classified by NBAR include:

- Class-based marking (the **set** command)
- Class-based weighted fair queueing (the **bandwidth** and **queue-limit** commands)
- Low latency queueing (the **priority** command)
- Traffic policing (the **police** command)
- Traffic shaping (the **shape** command)

Note The NBAR feature is used for classifying traffic by protocol. The other class-based QoS features determine how the classified traffic is forwarded and are documented separately from NBAR.

Furthermore, NBAR is not the only method of classifying network traffic so that QoS features can be applied to classified traffic.

For information on the class-based features that can be used to forward NBAR-classified traffic, see the individual feature modules for the particular class-based feature as well as the *Cisco IOS Quality of Service Solutions Configuration Guide*.

Many of the non-NBAR classification options for QoS are documented in the "Modular Quality of Service Command-Line Interface" section of the *Cisco IOS Quality of Service Solutions Configuration Guide*. These commands are configured using the **match** command in class map configuration mode.

NBAR introduces several new classification features that identify applications and protocols from Layer 4 through Layer 7:

- Statically assigned TCP and UDP port numbers
- Protocols that are non-UDP and non-TCP
- Dynamically assigned TCP and UDP port numbers. Classification of such applications requires stateful inspection; that is, the ability to discover the data connections to be classified by parsing the connections where the port assignments are made.
- Sub-port classification or classification based on deep packet inspection; that is, classification by looking deeper into the packet.

NBAR can classify static port protocols. Although access control lists (ACLs) can also be used for this purpose, NBAR is easier to configure and can provide classification statistics that are not available when using ACLs.

NBAR includes a Protocol Discovery feature that provides an easy way to discover application protocols that are transversing an interface. The Protocol Discovery feature discovers any protocol traffic supported by NBAR. Protocol Discovery maintains the following per-protocol statistics for enabled interfaces: total number of input and output packets and bytes, and input and output bit rates. The Protocol Discovery feature captures key statistics associated with each protocol in a network that can be used to define traffic classes and QoS policies for each traffic class.

For specific information about configuring NBAR and dNBAR, refer to the *Network-Based Application Recognition and Distributed Network-Based Application Recognition* feature documentation located at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/dtnbarad.htm

Configuring Hierarchical QoS on a SIP

Table 4-16 provides information about where the hierarchical QoS features for SPA interfaces are supported.

Table 4-16 Hierarchical QoS Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Hierarchical QoS for EoMPLS VCs	Supported for all SPAs in Cisco IOS Release 12.2(18)SXE and later, and in Cisco IOS Release 12.2(33)SRA.	Supported for all SPAs beginning in Cisco IOS Release 12.2(33)SRA.	Supported for all SPAs in Cisco IOS Release 12.2(18)SXF and later, and in Cisco IOS Release 12.2(33)SRA.
Hierarchical QoS—Tiered policy maps with parent policy using class-default only on the main interface.	Not applicable.	Supported for all SPAs in Cisco IOS Release 12.2(18)SXF and later.	Supported in Cisco IOS Release 12.2(18)SXF and later, and in Cisco IOS Release 12.2(33)SRA using **match vlan** command in parent policy.
Hierarchical QoS—Tiered policy maps with parent policy in user-defined or class-default classes on the main interface.	Supported for all SPAs in Cisco IOS Release 12.2(18)SXF and later, and in Cisco IOS Release 12.2(33)SRA.	Supported for all SPAs in Cisco IOS Release 12.2(33)SRA.	Not supported.

Configuring Hierarchical QoS with Tiered Policy Maps

Hierarchical QoS with tiered policy maps is a configuration where the actions associated with a class contain a queuing action (such as shaping) and a nested service policy, which in itself is a policy map with classes and actions. This hierarchy of the QoS policy map is then translated into a corresponding hierarchy of queues.

Hierarchical QoS with Tiered Policy Maps Configuration Guidelines

When configuring hierarchical QoS with tiered policy maps on a SIP, consider the following guidelines:

- For information about where hierarchical QoS with tiered policy maps is supported, see Table 4-16 on page 4-85.
- You can configure up to three levels of hierarchy within the policy maps.
- The parent policy map has the following restrictions on a main interface:
  - In Cisco IOS Release 12.2(18)SXF and later—Supports the shape queueing action in the default class (class-default) only.
  - In Cisco IOS Release 12.2(33)SRA—Supports VLAN or ACL matching, and shape or bandwidth queueing actions in any class, user-defined and class-default.
- When configuring hierarchical QoS for software-based EoMPLS on the Cisco 7600 SIP-600, if you configure **match input vlan** in the parent policy, then you can only configure **match qos-group** in the child policy.
- In hierarchical QoS, you cannot configure just a **set** command in the parent policy. The **set** command works only if you configure other commands in the policy.
- The child policy map supports shape, bandwidth, and WRED QoS features.

- With hierarchical QoS on a subinterface, the parent policy map supports hierarchical QoS using the **shape average** command as a queueing action in the default class (class-default) only.
- If you configure shaping at both the parent policy and the child policy, the traffic is shaped first according to the parameters defined in the parent policy, followed by the parameters of the child policy.
- If you configure service policies at the main interface, subinterface, and VC levels, then the policy applied at the VC level takes precedence over a policy at the interface.
- In a Frame Relay configuration, if you need to define service policies at the interface, subinterface, and PVC at the same time, then you can use a map class.
- For a POS subinterface with a Frame Relay PVC, a service policy can be applied either at the subinterface or at the PVC, but not both.
- Use a hierarchical policy if you want to achieve minimum bandwidth guarantees using CBWFQ with a map class. First, configure a parent policy to shape to the total bandwidth required (use the class-default in Cisco IOS Release 12.2(18)SXF, or a user-defined class beginning in Cisco IOS Release 12.2(33)SRA). Then, define a child policy using CBWFQ for the minimum bandwidth percentages.
- You can configure hierarchical QoS up to the following limits, according to the current Cisco IOS software limits:
  - Up to 1024 class maps
  - Up to 1024 policy maps
  - Up to 256 classes within a policy map

Configuring Hierarchical QoS for EoMPLS VCs

The Hierarchical Quality of Service (HQoS) for EoMPLS VCs feature extends support for hierarchical, parent and child relationships in QoS policy maps. This feature also provides EoMPLS per-VC QoS for point-to-point VCs.

The new feature adds the ability to match the virtual LAN (VLAN) IDs that were present on a packet when the packet was originally received by the router. It also supports the ability to match on a QoS group that is set to the same value of the IP precedence or 802.1P class of service (CoS) bits that are received on the incoming interface. This allows service providers to classify traffic easily for all or part of a particular EoMPLS network, as well as to preserve the customer's original differentiated services (DiffServ) QoS values.

In EoMPLS applications, the parent policy map typically specifies the maximum or the minimum bandwidth for a group of specific VCs in an EoMPLS network. Then child policy maps in the policy can implement a different bandwidth or perform other QoS operations (such as traffic shaping) on a subset of the selected VCs.

This feature enables service providers to provide more granular QoS services to their customers. It also gives service providers the ability to preserve customer IP precedence or CoS values in the network.

Note For information about where hierarchical QoS for EoMPLS VCs is supported, see Table 4-16 on page 4-85.

For more information about configuring hierarchical QoS for EoMPLS VCs, refer to the *Optical Services Module Configuration Note* located at the following URL:

http://www.cisco.com/application/pdf/en/US/guest/products/ps368/c1069/ccmigration_09186a008069bfcc.pdf

Configuring PFC QoS on a Cisco 7600 SIP-600

The Cisco 7600 SIP-600 supports most of the same QoS features as those supported by the Policy Feature Card on the Cisco 7600 series router.

This section describes those QoS features that have SIP-specific configuration guidelines. After you review the SIP-specific guidelines described in this document, then refer to the *Cisco 7600 Series Cisco IOS Software Configuration Guide, 12.2SX* located at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/software/122sx/swcg/index.htm

PFC QoS on a Cisco 7600 SIP-600 Configuration Guidelines

- Output policing is not supported.

Configuring Lawful Intercept on a Cisco 7600 SIP-400

This section describes configuring Lawful Intercept on a Cisco 7600 SIP-400. For initial configuration of the Lawful Intercept feature, see the *Cisco 7600 Lawful Intercept Configuration Guide* at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/products_configuration_guide_book09186a00807e0acb.html

To configure Lawful Intercept on a Cisco 7600 SIP-400, use the following commands:

Command	Purpose
Router(config)# **snmp-server view viewA ciscoTap2MIB included** Router(config)# **snmp-server view viewA ciscoIpTapMIB included**	Creates a view having access to the MIBS.
Router(config)# **snmp-server group groupA v3 auth read viewA write viewA notify viewA**	Creates a group having access to this view.
Router(config)# **snmp-server user user1 groupA v3 auth md5 cisco**	Creates a user who is a member of groupA.

Configuring Security ACLs on an Access Interface on a Cisco 7600 SIP-400

This section describes configuration of the SIP-specific ACL features on access interfaces. Before referring to any other ACL documentation for the platform or in the Cisco IOS software, use this section to determine SIP-specific ACL feature support and configuration guidelines.

An Access Control List (ACL) is a collection of ordered permit and deny statements, referred to as Access Control Entries (ACEs), which determine whether a particular packet will be forwarded or dropped. An ACL offers application layer awareness, providing operational staff with some flexibility in the level of isolation of a host. For instance, an ACL may be applied to enforce complete host isolation, denying all traffic to and from that particular host or, alternately, to just filter certain traffic flows, while permitting all others.

For additional details about ACL concepts and features in Cisco IOS Release 12.2, refer to the *Cisco IOS Security Configuration Guide, Release 12.2,* at the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a00800ca7c0.html

This section includes the following topics:

- Security ACL Configuration Guidelines, page 4-88
- Configuring Security ACL, page 4-88

Security ACL Configuration Guidelines

- Up to 100 unique ACLs are recommended per chassis, with a maximum of 24 ACEs per ACL for Security ACL.
- Up to one input ACL and one output ACL are recommended for all 8K subinterfaces on the SIP.
- Source and Destination IPv4 Address, Port Number, ToS/DSCP, Protocol type, and TCP flags can be specified in the ACEs. As of Cisco IOS Release 12.2(33)SRB, IPV6 is not supported.
- Template Security ACL is not supported as of Cisco IOS Release 12.2(33)SRB.
- Security ACLs are only supported on a Route Switch Processor 720 (RSP720) with a Cisco 7600 SIP-400.
- Standard, extended, and named ACLs are supported; other ACL types such as reflexive and time-based ACLs are not supported.

Configuring Security ACL

	Command or Action	Purpose
Step 1	Router(config)# **access-list** *access list number* **permit ip host** *ip address* **any**	Configures an access list.
Step 2	Router(config-int)# **interface gigabitethernet** *slot/subslot/port* **access**	Selects the gigabitethernet interface.
Step 3	Router(config-int)# **ip address** *address*	Specifies the IP address.
Step 4	Router(config-int)# **encapsulation dot1q** *vlan-id*	Enables traffic encapsulation. • *vlan-id*—Virtual LAN identifier; valid values are from 1 to 4094.

	Command or Action	Purpose
Step 5	Router(config-int)# **ip access-group** *access-list-number* **in**	Sets filtering method. • *access-list-number*—Number of an access list. This is a decimal number from 1 to 199 or 1300 to 2699. • **in**—Filters on inbound packets.
Step 6	Router(config-int)# **ip access-group** *access-list-number* **out**	Sets filtering method. • *access-list-number*—Number of an access list. This is a decimal number from 1 to 199 or 1300 to 2699. • **out**—Filters on outbound packets.

Verifying ACL Configuration

Use the following command to verify ACL configuration:

Command or Action	Purpose
Router# **show access-list** [*access-list-number* | *name*]	Displays access list configuration. • *access-list-number*—(Optional) Access list number to display. The range is 0 to 1199. The system displays all access lists by default. • *name*—(Optional) Name of the IP access list to display.

Configuring CoPP on the Cisco 7600 SIP-400

This section describes the configuration of Control Plane Policing (CoPP) on the Cisco 7600 SIP-400.

Because the majority of control plane processing is done on the CPU, a malicious user can attack a router by simply pumping control plane traffic to the router. On an unprotected router, this results in the CPU utilization nearing 100%, resource exhaustion, and the command line console being locked, intensifying the problem because the user is not able to apply any rectifying action on the router.

Using CoPP protects the control plane against these denial-of-service (DoS) attacks, ensuring routing stability, reachability, and packet delivery by providing filtering and rate-limiting capabilities for control plane packets.

For additional information regarding DoS and CoPP, refer to the *Cisco 7600 Series Router Cisco IOS Software Configuration Guide*.

This section contains the following topics:

- Configuring Per-Subscriber/Per-Protocol CoPP on Access Interfaces on a Cisco 7600 SIP-400, page 4-90
- Configuring Per-Subinterface CoPP on Access Interfaces on a Cisco 7600 SIP-400, page 4-91

Configuring Per-Subscriber/Per-Protocol CoPP on Access Interfaces on a Cisco 7600 SIP-400

This section describes the configuration of Per-Subscriber/Per-Protocol CoPP on a Cisco 7600 SIP-400.

Per-Subscriber/Per-Protocol CoPP Configuration Guidelines

- The Cisco 7600 CoPP feature is supported with a Route Switch Processor 720 (RSP720) and Cisco 7600 SIP-400 combination only.
- When enabling the RP-based aggregate CoPP functionality, the required class maps should be configured for each of the protocol-matching criteria. The CoPP policy maps should be created for all the protocols that need to be policed.
- Once the router processor decides to install a rate limiter on an interface, there will be a delay for actually installing the rate limiter on the Cisco 7600 SIP-400. During this interval, it is possible that the aggregate rate limiter would start dropping good user packets, if the per-interface rates are not chosen carefully. For example, consider that there are 10 interfaces and 100 pps is used as the aggregate rate and 15 pps as the per-interface rate. If there are seven attacks on the router at a time, the aggregate limit would be exceeded and user traffic would be affected.
- As of Cisco IOS Release 12.2(33)SRB, the CoPP Per-subscriber/Per-Protocol feature is only supported for DHCP, ARP, and ICMP protocols. DHCP and ARP policing are performed on the SPA, while ICMP policing is performed at the router processor level.

To configure Per-Subscriber/Per-Protocol CoPP support, use the following commands:

Command or Action	Purpose
Router(config)# **class-map arp-peruser**	Creates a class map for ARP.
Router(config-cmap)# **match protocol arp**	Matches ARP traffic.
Router(config-cmap)# **match subscriber access**	Defines the class map for access interfaces.
Router(config)# **class-map dhcp-peruser**	Creates a class map for DHCP.
Router(config-cmap)# **match protocol dhcp**	Configures the match criterion for a DHCP class map.
Router(config-cmap) **match subscriber access**	Defines the class map for access interfaces.
Router(config)# **policy-map copp-peruser**	Specifies CoPP as the policy map.
Router(config-pmap)# **class arp-peruser**	Creates an ARP peruser class.
Router(config-pmap-c)# **police rate** *units* **pps burst** *burst-in-packets* **packets**	Specifies the burst rate. • *units*—Rate at which traffic is policed in packets per second. Valid values are 1 to 2000000 pps. • *burst-in-packets*—(Optional) Specifies the burst rate that is used for policing traffic. Valid values are 1 to 512000 packets.
Router(config-pmap-c)# **class dhcp-peruser**	Creates a DHCP peruser class.

Command or Action	Purpose
Router(config-pmap-c)# **police rate** *units* **pps burst** *burst-in-packets* **packets**	Specifies the burst rate. • *units*—Rate at which traffic is policed in packets per second. Valid values are 1 to 2000000 pps. • *burst-in-packets*—(Optional) Specifies the burst rate that is used for policing traffic. Valid values are 1 to 512000 packets.
Router(config)# **control-plane user-type access**	Applies the policy on control-plane-user interface.
Router(config-cp-user)# **service-policy input copp-peruser**	Configures the per-user policy map.
Router(config)# **platform copp observation-period** *time*	Configures the observation window. • *time*—Amount of time in minutes.
Router# **platform copp** *interface* **arp off**	Clears a per-subinterface rate limiter for ARP on an interface. • *interface*—Defines interface.

Verifying Per-Subscriber/Per-Protocol CoPP

To verify Per-Subscriber/Per-Protocol CoPP configuration, use the following commands:

Command or Action	Purpose
Router# **show platform copp rate-limit** [*arp* | *dhcp* | *all*]	Displays configuration settings. • *arp*—Displays ARP information. • *dhcp*—Displays DHCP information. • *all*—Displays ARP and DHCP information.
Router# **show policy-map** *policy-map-name*	Verifies that packets match the desired class. • *policy-map-name*—(Optional) Name of the policy map.

Configuring Per-Subinterface CoPP on Access Interfaces on a Cisco 7600 SIP-400

This section describes the configuration of Per-Subinterface CoPP on a Cisco 7600 SIP-400.

Per-Subinterface CoPP Configuration Guidelines

This section describes guidelines to consider when configuring Per-Subinterface CoPP.

- Per-Subinterface CoPP is supported on Cisco 7600 series routers with Supervisor 720, SIP-400, and Ethernet SPAs.
- The following packet types can be rate-limited on the SIP-400:
  - DHCP packets
  - ARP packets

- ATM OAM packets
- Ethernet OAM packets
- PPPoE discovery packets

Note DHCP and ARP packets are supported in Cisco IOS Release 12.2(33)SRB and later. ATM OAM, Ethernet OAM, and PPPoE discovery packets are supported in Cisco IOS Release 12.2(33)SRC and later.

- If there is a normal QoS policy installed on an interface, the SIP-400 first applies the QoS policy, then the Security ACL, then the CoPP rate limiter on a packet.
- During a switchover, all dynamic rate limiters on the router are turned off.
- During online insertion and removal (OIR) of a line card, the rate limiters on the interfaces are reset.

Configuring Per-Subinterface CoPP

To configure Per-Subinterface CoPP support, use the following commands:

Command or Action	Purpose
Router(config)# **class-map** *class-map-name*	Creates a class map for the packet protocol.
Router(config-cmap)# **match protocol** *protocol-name* [**arp** | **dhcp** | **atm-oam** | **ethernet-oam** | **pppoe-discovery]**	Matches packet protocol traffic.
Router(config-cmap)# **match subscriber access**	Defines the class map for access interfaces.
Router(config)# **policy-map** *policy-map-name*	Specifies CoPP as the policy map.
Router(config-pmap)# **class** *class-map-name*	Creates a class map for the packet protocol.
Router(config-pmap-c)# **police rate** *units* [**pps burst** *burst-in-packets* **packets** | **bps burst** *burst-in-bytes* **bytes**]	Specifies the burst rate. • *units*—Rate at which traffic is policed in packets per second. Valid values are 1 to 2000000. • *burst-in-packets*—(Optional) Specifies the burst rate (in packets per second) that is used for policing traffic. Valid values are 1 to 512000 packets. • burst-in-bytes—(Optional) Specifies the burst rate (in bytes per second) that is used for policing traffic. Valid values are 100 to 1000 bytes.
Router(config)# **control-plane user-type access**	Applies the policy on the control-plane user interface.
Router(config-cp-user)# **service-policy input** *policy-map-name*	Configures the policy map.

Command or Action	Purpose
Router(config)# **platform copp observation-period** *time*	Configures the observation window. • *time*—Amount of time in minutes.
Router# **platform copp** *interface protocol-name* **off**	Clears a per-subinterface limiter for the packet protocol on an interface. • *interface*—Defines the interface. • *protocol-name*—Defines the packet protocol.

Verifying Per-Subinterface CoPP

To verify Per-Subinterface CoPP configuration, use the following commands:

Command or Action	Purpose
Router# **show platform copp rate-limit** *protocol-name* [**arp** | **dhcp** | **atm-oam** | **ethernet-oam** | **pppoe-discovery** | **all**]	Displays configuration settings for the selected packet protocol or all protocols.
Router# **show platform np copp** [*ifnum*] [**detail**]	Displays debug information for a given session or for all sessions. • *ifnum*—Identifies a specific session ID. • **detail**—Shows full rate-limiting values.

Configuring IGMP Snooping on a SIP-200

IGMP snooping constrains the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN router to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the router receives an IGMP report from a host for a particular multicast group, the router adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.

The multicast router sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The router creates one entry per VLAN in the IGMP snooping IP multicast forwarding table for each group from which it receives an IGMP join request.

For more information and configuration instructions, see the *Cisco 7600 Series Router IOS Software Configuration Guide, Release 12.2SR*.

Configuring ACFC and PFC Support on Multilink Interfaces

About ACFC and PFC

Using the Address and Control Field Compression (ACFC) and PPP Protocol Field Compression (PFC) Support on Multilink Interfaces feature, you can control the negotiation and application of the Link Control Protocol (LCP) configuration options for ACFC and PFC.

If ACFC is negotiated during Point-to-Point Protocol (PPP) negotiation, Cisco routers may omit the High-Level Data Link Control (HDLC) header on links using HDLC encapsulation. IF PFC is negotiated during PPP negotiation, Cisco routers may compress the PPP protocol field from two bytes to one byte.

The PPP commands described in this section provide options to control PPP negotiation, allowing the HDLC framing and the protocol field to remain uncompressed. These commands allow the system administrator to control when PPP negotiates the ACFC and PFC options during initial LCP negotiations and how the results of the PPP negotiation are applied.

Note Address and control field compression is only applicable to links that use PPP in HDLC-like framing as described by RFC 1662.

Restrictions and Usage Guidelines

ACFC and PFC should be configured with the link shut down.

Note When Multilink PPP is configured in hardware, ACFC and PFC are active only when all links in the bundle have ACFC and PFC configured.

Using ACFC and PFC can result in gains in effective bandwidth because they reduce the amount of framing overhead for each packet. However, using ACFC or PFC changes the alignment of the network data in the frame, which in turn can impair the switching efficiency of the packets both at the local and remote ends of the connection. For these reasons, it is generally recommended that ACFC and PFC not be enabled without carefully considering the potential results.

ACFC and PFC options are supported only when the serial interfaces are multilink member interfaces.

ACFC and PFC configured on MLP interfaces do not have any effect during PPP negotiation or during packet transmission.

Supported Platforms

SIP-200/SPA

This feature is supported on SIP-200 for the following SPAs:

- 2-Port and 4-Port Channelized T3 SPA
- 8-Port Channelized T1/E1 SPA
- 1-Port Channelized OC3/STM-1 SPA

Configuring ACFC and PFC Support

The following sections list the configuration tasks for ACFC and PFC handling.

Configuring ACFC Support

To configure ACFC support, perform the following tasks in interface configuration mode:

	Command	Purpose
Step 1	Router> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	Router# **configure terminal**	Enables global configuration mode.
Step 3	Router(config)# **interface serial** *slot***/***subslot***/***port***:***channel-group*	Selects the interface to configure. • *slot***/***subslot***/***port***:***channel-group*—Specifies the location of the interface.
Step 4	Router(config-if)# **shutdown**	Shuts down the interface.
Step 5	Router(config-if)# **ppp acfc remote** {**apply** | **reject** | **ignore**}	Configures how the router handles the ACFC option in configuration requests received from a remote peer. • **apply**—ACFC options are accepted and ACFC may be performed on frames sent to the remote peer. • **reject**—ACFC options are explicitly ignored. • **ignore**—ACFC options are accepted, but ACFC is not performed on frames sent to the remote peer.
Step 6	Router(config-if)# **ppp acfc local** {**request** | **forbid**}	Configures how the router handles ACFC in its outbound configuration requests. • **request**—The ACFC option is included in outbound configuration requests. • **forbid**—The ACFC option is not sent in outbound configuration requests, and requests from a remote peer to add the ACFC option are not accepted.
Step 7	Router(config-if)# **no shutdown**	Reenables the interface.

ACFC Configuration Example

The following example configures the interface to accept ACFC requests from a remote peer and perform ACFC on frames sent to the peer, and include the ACFC option in its outbound configuration in its outbound configuration requests:

```
Router> enable
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface serial 4/1/1/1:0
Router(config-if)# shutdown
```

```
Router(config-if)# ppp acfc remote apply
Router(config-if)# ppp acfc local request
Router(config-if)# no shutdown
```

Configuring PFC Support

To configure PFC support, perform the following tasks in interface configuration mode:

:

	Command	Purpose
Step 1	Router> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	Router# **configure terminal**	Enables global configuration mode.
Step 3	Router(config)# **interface serial** *slot***/***subslot***/***port***:***channel-group*	Selects the interface to configure. • *slot***/***subslot***/***port***:***channel-group*—Specifies the location of the interface.
Step 4	Router(config-if)# **shutdown**	Shuts down the interface
Step 5	Router(config-if)# **ppp pfc remote** {**apply** | **reject** | **ignore**}	Configures how the router handles the PFC option in configuration requests received from a remote peer. • **apply**—PFC options are accepted and PFC may be performed on frames sent to the remote peer. • **reject**—PFC options are explicitly ignored. • **ignore**—PFC options are accepted, but PFC is not performed on frames sent to the remote peer.
Step 6	Router(config-if)# **ppp pfc local** {**request** | **forbid**}	Configures how the router handles PFC in its outbound configuration requests. • **request**—The PFC option is included in outbound configuration requests. • **forbid**—The PFC option is not sent in outbound configuration requests, and requests from a remote peer to add the PFC option are not accepted.
Step 7	Router(config-if)# **no shutdown**	Reenables the interface.

PFC Configuration Example

The following example configures the interface to explicitly ignore the PFC option received from a remote peer, and exclude the PFC option from its outbound configuration requests and reject any request from a remote peer to add the PFC option:

```
Router> enable
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface serial 4/1/1/1:0
Router(config-if)# shutdown
Router(config-if)# ppp pfc remote reject
Router(config-if)# ppp pfc local forbid
Router(config-if)# no shutdown
```

Configuring PPPoEoE on a Cisco 7600 SIP-400

Point-to-Point Protocol (PPP) provides a standard method of communicating to peers over a point-to-point link. An Ethernet link provides multipoint communication between multiple peers. PPP over Ethernet (PPPoE) allows point-to-point communication across multipoint Ethernet links.

The PPPoE over Ethernet interface (PPPoEoE) enables the Cisco 7600 series router with Cisco 7600 SIP-400 to terminate Ethernet PPP sessions over Ethernet links. The PPPoE over IEEE 802.1Q VLANs feature enables the router to terminate Ethernet PPP sessions across VLAN links. IEEE 802.1Q encapsulation is used to interconnect a VLAN-capable router with another VLAN-capable networking device. The packets on the 802.1Q link contain a standard Ethernet frame and the VLAN information associated with that frame.

Supported Features

PPPoEoE on the Cisco 7600 SIP-400 supports the following features:

- PPPoE discovery packets (rate-limited), PPPoE PPP control packets, and PPPoE PPP IP data packets provide a per-user session on an Ethernet interface.
- PPPoE is supported on main interfaces, 802.1Q and QinQ access interfaces, and VLAN ranges (802.1Q ranges and QinQ inner ranges).
- 8 K PPPoE sessions are supported.
- PPPoE and IP sessions can be configured on the same subinterface.

Limitations and Restrictions

PPPoEoE on the Cisco 7600 SIP-400 has the following limitations and restrictions:

- PPP over ATM (PPPoA) is not supported.
- Tunneling of PPPoE sessions (Level 2 Tunneling Protocol) is not supported.
- Ambiguous VLANs and a range of VLANs for IP session interfaces are not supported. However, a range of VLANs is supported for PPPoE-configured interfaces.
- Negotiated maximum transmission unit (MTU) value can only be 1492 or 1500 bytes.
- If the **ip tcp adjust-mss** command is used, the only value supported is 1468.
- PPPoE can only be configured on subinterfaces using the **access** keyword.

Configuration Tasks for PPPoE over Ethernet

To configure PPPoE over Ethernet, perform the following tasks:

- Configuring a Virtual Template Interface, page 4-98
- Creating an Ethernet Interface and Enabling PPPoE, page 4-99
- Configuring PPPoE in a BBA Group, page 4-99
- Configuring PPPoE over 802.1Q VLANs on a Cisco 7600 SIP-400, page 4-100

Configuring a Virtual Template Interface

Configure a virtual template before you configure PPPoE on an Ethernet interface. The virtual template interface is a logical entity that is applied dynamically as needed to an incoming PPP session request. To create and configure a virtual template interface, enter the following commands beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface virtual-template** *number*	Creates a virtual template interface and enters interface configuration mode.
Step 2	Router(config-if)# **ip unnumbered ethernet** *number*	Enables IP without assigning a specific IP address on the LAN.
Step 3	Router(config-if)# **mtu** *bytes*	(Optional) Sets the maximum MTU size for the interface. **Note** MTU size can be set only to 1492 or 1500.
Step 4	Router(config-if)# **ppp authentication chap**	Enables PPP authentication on the virtual template interface.
Step 5	Router(config-if)# **ppp ipcp ip address required**	Required for legacy dial-up and DSL networks. Prevents a PPP session from being set up with 0.0.0.0 remote ip address.

The following example shows the configuration of a virtual template interface:

```
Router(config)# interface virtual-template 1
Router(config-if)# ip unnumbered ethernet 21
Router(config-if)# no peer default ip address
Router(config-if)# ppp authentication chap
Router(config-if)# ppp authorization vpn1
Router(config-if)# ppp accounting vpn1
```

Note The PPP commands shown in these examples are typical of virtual template configurations. Not all PPP commands are required. Refer to the PPP documentation for more information.

Monitoring and Maintaining a Virtual Access Interface

When a virtual template interface is applied dynamically to an incoming user session, a virtual access interface (VAI) is created. You cannot use the command line interface (CLI) to directly create or configure a VAI, but you can display and clear the VAI by using the following commands in privileged EXEC mode:

Command or Action	Purpose
Router# **show interfaces virtual-access** *number* [**configuration**]	Displays the configuration of the active VAI that was created using a virtual template interface. The **configuration** keyword restricts output to configuration information.
Router# **clear interface virtual-access** *number*	Tears down the live sessions and frees the memory for other client users.

The following example shows how to display the active VAI configuration:

```
Router# show interfaces virtual-access 1.1 configuration
!
interface virtual-access1.1
    if vrf forwarding vrf-1
    ip unnumbered Loopback1
    no ip proxy-arp
    peer default ip address pool vrf-1
    ppp authentication chap
    end
```

Note Virtual-access 1.1 is a PPPoE subinterface.

The following example shows how to clear a live session:

```
Router# clear interface virtual-access 1.1
Router#
```

Creating an Ethernet Interface and Enabling PPPoE

To create an Ethernet interface and enable PPPoE on it, enter the following commands beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface gigabitethernet** *number*	Creates an Ethernet interface and enters interface configuration mode.
Step 2	Router(config-if)# **protocol pppoe group** *group-name*	Enables PPPoE and allows PPPoE sessions to be created through that interface.

Configuring PPPoE in a BBA Group

Note Cisco IOS Release 12.2(33)SRC does not support the configuration of BBA groups using RADIUS. You must configure BBA groups manually.

To configure a broadband aggregation (BBA) group for PPPoE and link it to the appropriate virtual template interface, enter the following commands beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **bba-group pppoe** *name*	Configures a BBA group to be used to establish PPPoE sessions. *name* identifies the BBA group. You can have multiple BBA groups.
Step 2	Router(config-bba)# **virtual-template** *template-number*	Specifies the virtual template interface to use to clone VAIs.
Step 3	Router(config-bba)# **pppoe limit per-mac** *per-mac-limit*	(Optional) Specifies the maximum number of sessions per MAC address for each PPPoE port that uses the group.

	Command or Action	Purpose
Step 4	Router(config-bba)# **pppoe limit max-sessions** *number*	(Optional) Specifies the maximum number of PPPoE sessions that can be terminated on this router from all interfaces.
Step 5	Router(config-bba)# **pppoe limit per-vc** *per-vc-limit*	(Optional) Specifies the maximum number of PPPoE sessions for each VC that uses the group.
Step 6	Router(config-bba)# **exit**	Returns to global configuration mode.
Step 7	Router(config)# **interface** *type number* **access**	Specifies the type of interface to which you want to attach the BBA group and enters interface configuration mode. **Note** The access keyword is required on subinterfaces, but must not be used for main interfaces.
Step 8	Router(config-if)# **encapsulation dot1q** *vlan-id*	Enables IEEE 802.1Q encapsulation of traffic on a specified subinterface in a VLAN. Specify the VLAN identifier. **Note** This step is required only for 802.1Q and QinQ interfaces.
Step 9	Router(config-if)# **pppoe enable group** *group-name*	Attaches the BBA group to the VLAN.

Configuring PPPoE over 802.1Q VLANs on a Cisco 7600 SIP-400

PPPoE over IEEE 802.1Q VLANs enables the Cisco 7600 series router with the SIP-400 to support PPPoE over IEEE 802.1Q encapsulated VLAN interfaces. IEEE 802.1Q encapsulation is used to interconnect a VLAN-capable router with another VLAN-capable networking device. The packets on the 802.1Q link contain a standard Ethernet frame and the VLAN information associated with that frame.

Note PPPoE is disabled by default on a VLAN.

Configuring a Virtual Template

Before configuring PPPoE on an IEEE 802.1Q VLAN interface, configure a virtual template and a BBA group. See the "Configuring a Virtual Template Interface" section on page 4-98, and the "Configuring PPPoE in a BBA Group" section on page 4-99.

Creating an Ethernet IEEE 802.1Q Encapsulated Subinterface and Enabling PPPoE

To create an Ethernet 802.1Q interface and enable PPPoE on it, enter the following commands beginning in global configuration mode.

	Command or Action	Purpose
Step 1	Router(config)# **interface gigabitethernet** *slot/subslot/port.number* **access**	Creates a Gigabit Ethernet subinterface and enters subinterface configuration mode.

	Command or Action	Purpose
Step 2	Router(config-subif) # **encapsulation dot1q** *vlan-id* [**second-dot1q** *inner-vlan-id*]	Enables IEEE 802.1Q encapsulation on a specified subinterface in VLANs.
Step 3	Router(config-subif)# **pppoe enable group** *group-name*	Enables PPPoE and allows PPPoE sessions to be created through the specified subinterface.

Verifying PPPoE over Ethernet and IEEE 802.1Q VLAN

To verify PPPoEoE and IEEE 802.1Q VLAN, enter the following commands in privileged EXEC mode:

Command or Action	Purpose
Router# **show pppoe session all**	Displays PPPoE session information for each session ID.
Router# **show pppoe session packets**	Displays PPPoE session statistics.
Router# **show pppoe summary**	Displays PPPoE summary statistics.

Clearing PPPoE Sessions

To clear PPPoE sessions, enter the following commands in privileged EXEC mode:

Command or Action	Purpose
Router# **clear pppoe all**	Clears all PPPoE sessions.
Router# **clear pppoe interface**	Clears all PPPoE sessions on a physical interface or subinterface.
Router# **clear pppoe rmac**	Clears PPPoE sessions from a client host MAC address.
Router# **pppoe interface** *interface* **vlan** *vlan-number*	Clears sessions on a per-VLAN basis in ambiguous VLAN cases.

Configuring Source IPv4 and Source MAC Address Binding on the SIP-400

The Source IPv4 and Source MAC Address Binding feature is used in conjunction with the DHCP Authorized ARP and Secure ARP features to provide a check of the source IPv4 and source MAC address binding information before a packet can proceed to a higher level of processing. If the binding information does not exist, the packet is dropped.

Configuration Guidelines

When configuring source IPv4 and source MAC address binding, follow these guidelines:

- Supports access subinterfaces on the Cisco 7600 series routers in DHCP and non-DHCP environments.

Note Static entry of the MAC and IP address is required in a non-DHCP environment.

- Supports IPv4 unicast packets only.
- Supports Ethernet interfaces, subinterfaces, and routed Switched Virtual Interfaces (SVIs).
- Supports interface/subinterface and intelligent edge (iEdge) IP sessions.
- Supports up to 128000 IPv4 and MAC address bindings (subscriber entries) for the Cisco 7600 series router, and 8000 MAC address subscriber entries for each Cisco 7600 SIP-400.
- This feature is recommended primarily for access-facing interfaces and subinterfaces.
- Supports Cisco 7600 series router with RSP720, SUP720, or SUP 32.
- Supports on Cisco 7600 SIP-400 for the following Ethernet SPAs:
  - 2-Port Gigabit Ethernet SPA
  - 5-Port Gigabit Ethernet SPA
  - 10-Port Gigabit Ethernet SPA
- Supports only Ethernet and Ethernet logical interfaces. This feature can be supported on other interfaces provided they have Ethernet encapsulations underneath their primary encapsulation (for example, RBE or routed bridged PVC or EVC).
- If you are using EVC, this feature must be configured for bridge domain.

Restrictions

When configuring source IPv4 and source MAC address binding, note these restrictions:

- This feature cannot be used if multiple clients are using the same MAC address and they are on the same logical interfaces (VLAN).
- This feature does not support native LAN cards on the Cisco 7600 series router.
- This feature supports only one EVC per SVI.

Configuring Source IPv4 and Source MAC Address Binding

To configure this feature, perform the following tasks:

- Securing ARP Table Entries to DHCP Leases, page 4-102
- Configuring the Interfaces for Source IPv4 and Source MAC Address Binding, page 4-103
- Configuring DHCP Authorized ARP, page 4-104
- Showing the Number of Dropped Packets, page 4-105

Securing ARP Table Entries to DHCP Leases

This task describes how to secure ARP table entries to DHCP leases, starting in global configuration mode.

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.

	Command	Purpose
Step 2	Router(config)# **ip dhcp pool** *pool-name*	Configures a DHCP address pool and enters DHCP pool configuration mode. *pool-name*—Name of the pool. Can either be a symbolic string or an integer.
Step 3	Router(dhcp-config)# **network** *network-number*	Configures the network number and mask for a DHCP address pool. *network-number*—IP address of the primary DHCP address pool. **Note** Use the **network** command to configure the Cisco 7600 series router as a DHCP server. Otherwise, the Cisco 7600 acts as a DHCP relay agent and gets the address from an outside server.
Step 4	Router(dhcp-config)# **update arp**	Secures insecure ARP table entries to the corresponding DHCP leases.
Step 5	Router(dhcp-config)# **exit**	Exits DHCP pool configuration mode.

Example:

```
Router# configure terminal
Router(config)# ip dhcp pool tc10
Router(dhcp-config)# network 10.0.0.0 255.255.255.0
Router(dhcp-config)# update arp
Router(dhcp-config)# exit
```

Configuring the Interfaces for Source IPv4 and Source MAC Address Binding

This task describes how to enable source IPv4 and source MAC address binding in interface configuration mode.

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **interface vlan** *vlan-number*	Specifies interface and VLAN number and enters interface configuration mode. *vlan-number*—Range is from 1 to 4094. **Note** To configure a main interface, use the **interface** *type slot/subslot/port* command in global configuration mode.
Step 3	Router(config-if)# **ip address** *ip-address mask*	Sets an IP address for an interface. *ip-address*—IP address. *mask*—Mask for the associated subnet.
Step 4	Router(config-if)# **ip verify unicast source reachable-via rx l2-src**	Enables source IPv4 and source MAC address binding.
Step 5	Router(config-if)# **no shutdown**	Enables the interface.

Example:

```
Router# configure terminal
Router(config)# interface vlan 10
Router(config-if)# ip address 10.0.0.1 255.255.255.0
Router(config-if)# ip verify unicast source reachable-via rx l2-src
Router(config-if)# no shutdown
```

Configuring DHCP Authorized ARP

This task describes how to disable dynamic ARP learning on an interface, starting in interface configuration mode.

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **interface** *type slot/subslot/port*	Configures an interface type and enters interface configuration mode. *type slot/subslot/port*—Specifies the type and location of the interface.
Step 3	Router(config-if)# **arp authorized**	Disables dynamic ARP learning on an interface.
Step 4	Router(config-if)# **arp timeout** *seconds*	Configures how long an entry remains in the ARP cache. *seconds*—Time (in seconds) that an entry remains in the ARP cache. A value of 0 means that entries are never cleared from the cache.
Step 5	Router(config-if)# **service instance** *id* **ethernet**	Configures an Ethernet service instance on an interface and enters Ethernet service configuration mode. *id*—Integer in the range of 1 to 4294967295 that uniquely identifies a service instance on an interface.
Step 6	Router(config-if-srv)# **encapsulation dot1q** *vlan-id*	Defines the matching criteria to map 802.1Q frames ingress on an interface to the appropriate service instance. *vlan-id*—VLAN ID, an integer in the range 1 to 4094.
Step 7	Router(config-if-srv)# **rewrite ingress tag pop** {**1** | **2**} **symmetric**	Specifies the encapsulation adjustment to be performed on the frame ingress to the service instance. **pop** {**1** | **2**}—One or two tags are removed from the packet. **symmetric**—(Optional) Specifies tagging on the packets in the reverse direction (egress).
Step 8	Router(config-if-serv)# **bridge-domain** *bridge-id*	Binds the service instance to a bridge domain instance. *bridge-id*—Identifier for the bridge domain instance, an integer in the range of 1 to a platform-specific upper limit.
Step 9	Router(config-if-srv)# **no shutdown**	Enables the interface.
Step 10	Router(config-if-srv)# **end**	Ends the current configuration session and returns to privileged EXEC mode.

Example:

```
Router# configure terminal
Router(config)# interface gigabitethernet 8/0/1
Router(config-if)# arp authorized
Router(config-if)# arp timeout 60
```

```
Router(config-if)# service instance 101 ethernet
Router(config-if-srv)# encapsulation dot1q 101
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
Router(config-if-srv)# bridge-domain 10
Router(config-if-srv)# no shutdown
Router(config-if-srv)# end
```

Showing the Number of Dropped Packets

This task describes how to display the number of packets dropped when the source IPv4 and source MAC address binding check has failed.

	Command	Purpose
Step 1	Router# **attach** *slot-number*	Attaches to the SIP-400. *slot-number*—location of SIP-400.
Step 2	SIP-400-8# **show platform drops detail**	(Router prompt changes to SIP-400 prompt.) Shows statistics regarding dropped packets.

Example”

```
Router# attach 8
Entering CONSOLE for slot 8
Type "^C^C^C" to end this session

SIP-400-8# show platform drops detail

Global drops:
Drops for all interfaces:
 Gi8/0/0 ENP ifixp            16    Source masking (normal occurrence)
 Gi8/0/1 INP ifixp            3     BPDUs are not supported on this i/f
 Gi8/0/1 ENP ifixp            2008  Source masking (normal occurrence)
 Gi8/0/1 INP ifixp            2000  Src IP/MAC check failed
 Gi8/0/1 ENP ifixp            13    Source masking (normal occurrence)

SIP-400-8#
```

Resetting a SIP

To reset a SIP, use the following command in privileged EXEC configuration mode:

Command	Purpose
Router# **hw-module module** *slot* **reset**	Turns power off and on to the SIP in the specified slot, where: • *slot*—Specifies the chassis slot number where the SIP is installed.

Configuration Examples

This section includes the following examples for configuring SIPs installed in a Cisco 7600 series router:

- Layer 2 Interworking Configuration Examples, page 4-106
- MPLS Configuration Examples, page 4-108
- QoS Configuration Examples, page 4-108

Layer 2 Interworking Configuration Examples

This section includes the following Layer 2 interworking configuration examples:

- BCP in Trunk Mode Configuration Example, page 4-106
- BCP in Single-VLAN Mode Configuration Example, page 4-107

BCP in Trunk Mode Configuration Example

The following example shows how to configure BCP in trunk mode:

```
! Enter global configuration mode.
!
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address.
!
Router(config)# interface pos4/1/0
!
! Put the interface in Layer 2 mode for Layer 2 configuration.

Router(config-if)# switchport
%Please shut/no shut POS4/1/0 to bring up BCP
!
! When the switchport command is configured, the interface is automatically configured for
! trunk mode and nonegotiate status.
! Restart the interface to enable BCP.
!
Router(config-if)# shutdown
Router(config-if)# no shutdown
!
! Enable all VLANs for receiving and transmitting traffic on the trunk.
!
Router(config-if)# switchport trunk allowed vlan all
%Internal vlans not available for bridging:1006-1018,1021
```

The following example shows sample output from the **show running-config** command for this configuration. The **switchport mode trunk** and **switchport nonegotiate** commands are automatically NVgened when the **switchport** command is configured:

```
Router# show running-config interface pos4/1/0
Building configuration...
Current configuration : 191 bytes
!
interface POS4/1/0
switchport
switchport trunk allowed vlan all
switchport mode trunk
```

```
switchport nonegotiate
no ip address
encapsulation ppp
clock source internal
end
```

BCP in Single-VLAN Mode Configuration Example

The following example shows how to configure BCP in single-VLAN mode:

```
! Enter global configuration mode.
!
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address
!
Router(config)# interface pos4/1/0
!
! Disable IP processing on the interface. This is recommended for BCP interfaces.
!
Router(config-if)# no ip address
!
! Configure PPP encapsulation. You must configure PPP encapsulation before using the
! bridge-domain command.
!
Router(config-if)# encapsulation ppp
!
! Configure the bridging domain tag all Ethernet frames on the BCP link with the 802.1Q
! header.
!
Router(config-if)# bridge-domain 100 dot1q
%Please shut/no shut POS4/1/0 to bring up BCP
!
! Restart the interface to enable BCP.
!
Router(config-if)# shutdown
Router(config-if)# no shutdown
```

The following example shows sample output from the **show running-config** command for this configuration:

```
Router# show running-config interface pos4/1/0
Building configuration...
Current configuration : 122 bytes
!
interface POS4/1/0
no ip address
encapsulation ppp
bridge-domain 100 dot1q
clock source internal
end
```

The following example shows an example of the message that is sent if you attempt to configure the **bridge-domain** command without configuring PPP encapsulation:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface pos4/1/0
Router(config-if)# bridge-domain 100 dot1q
Must set encapsulation to PPP before using hw bridging over PPP
```

MPLS Configuration Examples

This section includes the following MPLS configuration examples:

Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Class-Based Tunnel Selection (CBTS) Configuration Example

The following example shows how to configure Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Class-Based Tunnel Selection (CBTS). Tunnel1, Tunnel2, and Tunnel3 are member tunnels, and Tunnel4 is the master tunnel.

```
Router(config)# interface Tunnel1
Router(config-if)# ip unnumbered loopback0
Router(config-if)# interface destination 24.1.1.1
Router(config-if)# tunnel mode mpls traffic-eng
Router(config-if)# tunnel mpls traffic-eng bandwidth sub-pool 30000
Router(config-if)# tunnel mpls traffic-eng exp 5

Router(config)# interface Tunnel2
Router(config-if)# ip unnumbered loopback0
Router(config-if)# interface destination 24.1.1.1
Router(config-if)# tunnel mode mpls traffic-eng
Router(config-if)# tunnel mpls traffic-eng bandwidth 50000
Router(config-if)# tunnel mpls traffic-eng exp 3 4

Router(config)# interface Tunnel3
Router(config-if)# ip unnumbered loopback0
Router(config-if)# interface destination 24.1.1.1
Router(config-if)# tunnel mode mpls traffic-eng
Router(config-if)# tunnel mpls traffic-eng bandwidth 10000
Router(config-if)# tunnel mpls traffic-eng exp default

Router(config)# interface Tunnel4
Router(config-if)# interface destination 24.1.1.1
Router(config-if)# tunnel mpls traffic-eng exp-bundle master
Router(config-if)# tunnel mpls traffic-eng exp-bundle member Tunnel1
Router(config-if)# tunnel mpls traffic-eng exp-bundle member Tunnel2
Router(config-if)# tunnel mpls traffic-eng exp-bundle member Tunnel3
Router(config-if)# tunnel mpls traffic-eng autoroute enable
```

QoS Configuration Examples

This section includes the following QoS configuration examples:

QoS with Multipoint Bridging Configuration Examples

The SIPs and SPAs support a subset of QoS features with MPB configurations.

- For ATM bridging, Frame Relay bridging, MPB, and BCP features on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, the following matching features are supported on bridged frames beginning in Cisco IOS Release 12.2(33)SRA:
  - Matching on ATM CLP bit
  - Matching on Frame Relay DE bit
  - Matching on Frame Relay DLCI
  - Matching on inner VLAN
  - Matching on inner CoS
  - Matching on IP DSCP (input interface only)
- For ATM bridging, Frame Relay bridging, MPB, and BCP features on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, the following marking features are supported on bridged frames beginning in Cisco IOS Release 12.2(33)SRA:
  - Set ATM CLP bit (output interface only)
  - Set Frame Relay DE bit (output interface only)
  - Set inner CoS
- For ATM bridging, Frame Relay bridging, MPB, and BCP features on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, the following marking features with policing are supported on bridged frames beginning in Cisco IOS Release 12.2(33)SRA:
  - Set inner CoS

For more information about configuring QoS on SIPs and SPAs, see the "Configuring QoS Features on a SIP" section on page 4-61.

This section includes the following QoS with MPB configuration examples:

Matching All Traffic on an Inner VLAN Tag with MPB on SIPs and SPAs on the Cisco 7600 Series Router Example

You can match traffic on an inner VLAN ID of a packet when you are using bridging features on a SPA. The following example shows configuration of a QoS class that filters all bridged traffic for VLAN 100 into a class named "vlan-inner-100." An output service policy is then applied to the SPA interface that bridges all outgoing traffic for the vlan-inner-100 class into VLAN 100.

```
! Configure the class maps with your matching criteria.
!
Router(config)# class-map match-all vlan-inner-100
Router(config-cmap)# match vlan inner 100
!
! Apply the service policy to an input or output bridged interface or VC.
!
Router(config)# interface atm3/0/0
Router(config-if)# pvc 100/100
Router(config-if-atm-vc)# bridge-domain 100 dot1q
```

```
Router(config-if-atm-vc)# service-policy output vlan-inner-100
Router(config-if)# end
```

Marking the Inner CoS Value with MPB on SIPs and SPAs on the Cisco 7600 Series Router Example

The following example shows configuration of a QoS class that filters all traffic matching on VLAN 100 into a class named "vlan-inner-100." The configuration shows the definition of a policy-map (also named "vlan-inner-100") that marks the inner CoS with a value of 3 for traffic in the vlan-inner-100 class. Since marking of the inner CoS value is only supported with bridging features, the configuration also shows the service policy being applied as an output policy to a serial SPA interface that bridges traffic into VLAN 100 using the **bridge-domain** command.

```
! Configure the class maps with your matching criteria.
!
Router(config)# class-map match-all vlan-inner-100
Router(config-cmap)# match vlan inner 100
Router(config-cmap)# exit
!
! Configure the policy map to mark all traffic in a class.
!
Router(config)# policy-map vlan-inner-100
Router(config-pmap)# class vlan-inner-100
Router(config-pmap-c)# set cos-inner 3
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
! Apply the service policy to an input or output bridged interface or VC.
!
Router(config)# interface serial3/0/0
Router(config-if)# no ip address
Router(config_if)# encapsulation ppp
Router(config-if)# bridge-domain 100 dot1q
Router(config-if)# service-policy output vlan-inner-100
Router(config-if)# shutdown
Router(config-if)# no shutdown
Router(config-if)# end
```

Configuring QoS Matching, Shaping, and Marking with MPB on SIPs and SPAs on the Cisco 7600 Series Router Example

The following example shows a complete QoS configuration of matching, shaping, and marking with MPB on SIPs and SPAs.

```
! Configure the class maps with your matching criteria.
! The following class maps configure matching on the inner VLAN ID.
!
Router(config)# class-map match-all vlan100
Router(config-cmap)# match vlan inner 100
Router(config-cmap)# exit
Router(config)# class-map match-all vlan200
Router(config-cmap)# match vlan inner 200
Router(config-cmap)# exit
Router(config)# class-map match-all vlan300
Router(config-cmap)# match vlan inner 300
Router(config-cmap)# exit
!
! The following class maps configure matching on the inner COS value.
!
Router(config)# class-map match-all cos0
Router(config-cmap)# match cos inner 0
Router(config-cmap)# exit
Router(config)# class-map match-all cos1
Router(config-cmap)# match cos inner 1
```

```
Router(config-cmap)# exit
Router(config)# class-map match-all cos2
Router(config-cmap)# match cos inner 2
Router(config-cmap)# exit
Router(config)# class-map match-all cos7
Router(config-cmap)# match cos inner 7
Router(config-cmap)# exit
!
! Configure a policy map for the defined classes.
! The following policies define shaping characteristics for classes
! on different VLANs
!
Router(config)# policy-map vlan100
Router(config-pmap)# class cos1
Router(config-pmap-c)# bandwidth percent 10
Router(config-pmap-c)# exit
Router(config-pmap)# class cos2
Router(config-pmap-c)# bandwidth percent 20
Router(config-pmap-c)# exit
Router(config-pmap)# class cos7
Router(config-pmap-c)#  percent 30
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# policy-map vlan200
Router(config-pmap)# class cos1
Router(config-pmap-c)# bandwidth percent 10
Router(config-pmap-c)# exit
Router(config-pmap)# class cos2
Router(config-pmap-c)# bandwidth percent 20
Router(config-pmap-c)# exit
Router(config-pmap)# class cos7
Router(config-pmap-c)#  percent 30
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
! The following policy map defines criteria for an output interface using MPB
!
Router(config)# policy-map egress_mpb
Router(config-pmap)# class vlan100
Router(config-pmap-c)# bandwidth percent 30
Router(config-pmap-c)# service-policy vlan100
Router(config-pmap-c)# exit
Router(config-pmap)# class vlan200
Router(config-pmap-c)# bandwidth percent 40
Router(config-pmap-c)# service-policy vlan200
!
! The following policy map defines criteria for an input interface using MPB
!
Router(config)# policy-map ingress_mpb
Router(config-pmap)# class vlan100
Router(config-pmap-c)# set cos-inner 5
Router(config-pmap-c)# exit
Router(config-pmap)# class vlan200
Router(config-pmap-c)# set cos-inner 3
!
! The following policy map defines criteria for an ATM output interface using MPB
! Note: You can only mark ATM CLP on an ATM output interface with MPB
!
Router(config)# policy-map atm_clp
Router(config-pmap)# class cos1
Router(config-pmap-c)# set atm-clp
Router(config-pmap-c)# exit
Router(config-pmap)# class cos2
Router(config-pmap-c)# set atm-clp
```

```
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
! Configure an interface for MPB and apply the service policies.
! The following example configures a POS interface in BCP trunk mode and applies two
! different service policies for the output and input traffic on the interface.
!
Router(config)# interface POS3/0/0
Router(config-if)# switchport
Router(config-if)# shutdown
Router(config-if)# no shutdown
Router(config-if)# switchport trunk allowed vlan 100,200,300
Router(config-if)# service-policy output egress_mpb
Router(config-if)# service-policy input ingress_mpb
!
! The following example configures an ATM interface with bridging on VLAN 100
! and applies a service policy for setting the ATM CLP for the output traffic.
!
Router(config)# interface ATM 4/1/0
Router(config-if)# pvc 1/100
Router(config-if-atm-vc)# bridge-domain 100
Router(config-if-atm-vc)# service-policy output atm-clp
```

Setting the Inner CoS Value as a Policing Action for SIPs and SPAs on the Cisco 7600 Series Router Example

The following example shows configuration of a QoS class that filters all traffic for virtual LAN (VLAN) 100 into a class named "vlan-inner-100," and establishes a traffic shaping policy for the vlan-inner-100 class. The service policy limits traffic to a CIR of 20 percent and a PIR of 40 percent, with a conform burst (bc) of 300 ms, and peak burst (be) of 400 ms, and sets the inner CoS value to 3. Because setting of the inner CoS value is only supported with bridging features, the configuration also shows the service policy being applied as an output policy for an ATM SPA interface permanent virtual circuit (PVC) that bridges traffic into VLAN 100 using the **bridge-domain** command.

```
! Configure the class maps with your matching criteria
!
Router(config)# class-map match-all vlan-inner-100
Router(config-cmap)# match vlan inner 100
Router(config-cmap)# exit
!
! Configure the policy map to police all traffic in a class and mark conforming traffic
! (marking traffic whose rate is less than the conform burst)
!
Router(config)# policy-map vlan-inner-100
Router(config-pmap-c)# police cir percent 20 bc 300 ms be 400 ms pir percent 40
conform-action set-cos-inner-transmit 3
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
! Apply the service policy to an input or output bridged interface or VC.
!
Router(config)# interface atm3/0/0
Router(config-if)# pvc 100/100
Router(config-if-atm-vc)# bridge-domain 100 dot1q
Router(config-if-atm-vc)# service-policy output vlan-inner-100
Router(config-if)# end
```

Hierarchical QoS with 2-Level Policy Map Configuration Examples

The following example shows configuration of hierarchical QoS that maps to two levels of hierarchical queues (you can configure up to three levels). The first-level policy (the parent policy) configures the aggregated data rate to be shaped to 1 Mbps for the class-default class. The second-level policy (the child policy) configures the traffic in User-A class for 40 percent of the bandwidth and traffic in User-B class for 60 percent of the bandwidth.

Because this example shows the parent policy applying to the class-default class, it is supported in Cisco IOS Release 12.2(33)SXF and later, as well as in Cisco IOS Release 12.2(33)SRA.

```
! Configure the class maps with your matching criteria
!
Router(config)# class-map match-any User-A
Router(config-cmap)# match access-group A
Router(config-cmap)# exit
Router(config)# class-map match-any User-B
Router(config-cmap)# match access-group B
Router(config-cmap)# exit
!
! Configure the parent policy for class-default to shape
! all traffic in that class and apply a second-level policy.
!
Router(config)# policy-map parent
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape 1000000
Router(config-pmap-c)# service-policy child
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
! Configure the child policy to allocate different percentages of
! bandwidth by class.
!
Router(config)# policy-map Child
Router(config-pmap)# class User-A
Router(config-pmap-c)# bandwidth percent 40
Router(config-pmap-c)# exit
Router(config-pmap)# class User-B
Router(config-pmap-c)# bandwidth percent 60
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
! Apply the parent service policy to an input or output interface.
!
Router(config)# interface GigabitEthernet 2/0/0
Router(config-if)# service-policy output parent
```

The following example shows configuration of hierarchical QoS that maps to two levels of hierarchical queues, where the parent policy configures average traffic shaping rates on both user-defined classes as well as the class-default class, which is supported beginning in Cisco IOS Release 12.2(33)SRA. This configuration does not show the corresponding class map configuration, which also are required to support these policy maps.

```
! Configure the parent policy for user-defined and class-default classes to shape
! traffic in those classes and apply a second-level policy.
!
Router(config)# policy-map parent
Router(config-pmap)# class input-vlan100
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service-policy child-pm
Router(config-pmap-c)# exit
Router(config-pmap)# class input-vlan200
```

```
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service-policy child-pm
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape average 200000
Router(config-pmap-c)# service-policy child-pm
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
! Configure the child policy to allocate different percentages of
! bandwidth by class.
!
Router(config)# policy-map child-pm
Router(config-pmap)# class cos0
Router(config-pmap-c)# bandwidth percent 10
Router(config-pmap-c)# exit
Router(config-pmap)# class cos1
Router(config-pmap-c)# bandwidth percent 10
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
! Apply the parent service policy to an input or output interface.
!
Router(config)# interface gigabitethernet 2/0/0
Router(config-if)# service-policy output parent-pm
```

CHAPTER 5

Troubleshooting the SIPs and SSC

This chapter describes techniques that you can use to troubleshoot the operation of your SIPs.

It includes the following sections:

The first section provides information about basic interface troubleshooting. If you are having a problem with your SPA, use the steps in the "Using the Cisco IOS Event Tracer to Troubleshoot Problems" section to begin your investigation of a possible interface configuration problem.

To perform more advanced troubleshooting, see the other sections in this chapter.

General Troubleshooting Information

This section describes general information for troubleshooting SIPs, SSCs, and SPAs. It includes the following sections:

Interpreting Console Error Messages

To view the explanations and recommended actions for Cisco 7600 series router error messages, including messages related to Cisco 7600 series router SIPs and SSCs, refer to the following documents:

- *Cisco 7600 Series Cisco IOS System Message Guide, 12.2SX* (for error messages in Release 12.2SX)
- *System Error Messages for Cisco IOS Release 12.2S* (for error messages in Release 12.2S)

System error messages are organized in the documentation according to the particular system facility that produces the messages. The SIP and SSC error messages use the following facility names:

- Cisco 7600 SIP-200—C7600_SIP200
- Cisco 7600 SIP-400—SIP400

- Cisco 7600 SIP-600—SIP600
- Cisco 7600 SSC-400—C7600_SSC400

Using debug Commands

Along with the other **debug** commands supported on the Cisco 7600 series router, you can obtain specific debug information for SIPs and SSCs on the Cisco 7600 series router using the **debug hw-module** privileged EXEC command.

The **debug hw-module** command is intended for use by Cisco Systems technical support personnel.

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

For more information about other debug commands that can be used on a Cisco 7600 series router, refer to the *Cisco 7600 Series Cisco IOS Command Reference, 12.2 SX* and to the *Cisco IOS Debug Command Reference, Release 12.2 SR*.

Using show Commands

There are several **show** commands that you can use to monitor and troubleshoot the SIPs and SSCs on the Cisco 7600 series router. This chapter describes using the **show hw-module slot** command to perform troubleshooting of your SPA.

For more information about **show** commands to verify and monitor SIPs and SSCs, see the following chapters of this guide:

- Chapter 4, "Configuring the SIPs and SSC"

Using the Cisco IOS Event Tracer to Troubleshoot Problems

Note This feature is intended for use as a software diagnostic tool and should be configured only under the direction of a Cisco Technical Assistance Center (TAC) representative.

The Event Tracer feature provides a binary trace facility for troubleshooting Cisco IOS software. This feature gives Cisco service representatives additional insight into the operation of the Cisco IOS software and can be useful in helping to diagnose problems in the unlikely event of an operating system malfunction or, in the case of redundant systems, Route Processor switchover.

Event tracing works by reading informational messages from specific Cisco IOS software subsystem components that have been preprogrammed to work with event tracing, and by logging messages from those components into system memory. Trace messages stored in memory can be displayed on the screen or saved to a file for later analysis.

The SPAs currently support the "spa" component to trace SPA OIR-related events.

For more information about using the Event Tracer feature, refer to the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide09186a0080087164.html

Troubleshooting Oversubscription on the Cisco 7600 SIP-400

As of Cisco IOS Release 12.2(18)SXF, when using the Cisco 7600 SIP-400 with the 2-Port Gigabit Ethernet SPA or the 1-Port OC-48c/STM-16 ATM SPA, consider the following oversubscription guidelines:

- The Cisco 7600 SIP-400 only supports installation of one 1-Port OC-48c/STM-16 ATM SPA without any other SPAs installed in the SIP.
- The Cisco 7600 SIP-400 supports installation of up to two 2-Port Gigabit Ethernet SPAs without any other SPAs installed in the SIP.
- The Cisco 7600 SIP-400 supports installation of any combination of OC-3 or OC-12 POS or ATM SPAs, up to a combined ingress bandwidth of OC-48 rates.
- The Cisco 7600 SIP-400 supports installation of any combination of OC-3 or OC-12 POS or ATM SPAs up to a combined ingress bandwidth of OC-24 rates, when installed with a single 2-Port Gigabit Ethernet SPA.

Configurations on the Cisco 7600 SIP-400 with an unsupported aggregate SPA bandwidth greater than OC-48 rates generates the following error message:

```
SLOT 3: 00:00:05: %SIPSPA-4-MAX_BANDWIDTH: Total SPA bandwidth exceeds line card capacity
of 2488 Mbps
```

Preparing for Online Insertion and Removal of SIPs, SSCs, and SPAs

The Cisco 7600 series router supports online insertion and removal (OIR) of the SPA interface processor (SIP) or SPA services card (SSC), in addition to each of the shared port adapters (SPAs). Therefore, you can remove a SIP or SSC with its SPAs still intact, or you can remove a SPA independently from the SIP or SSC, leaving the SIP or SSC installed in the router.

This section includes the following topics on OIR support:

- Preparing for Online Removal of a SIP or SSC, page 5-4
- Verifying Deactivation and Activation of a SIP or SSC, page 5-5
- Preparing for Online Removal of a SPA, page 5-6
- Verifying Deactivation and Activation of a SPA, page 5-7
- Deactivation and Activation Configuration Examples, page 5-8

Note For simplicity, any reference to "SIP" in this section also applies to the SSC.

Preparing for Online Removal of a SIP or SSC

The Cisco 7600 series router supports OIR of the SIP and the SSC. To do this, you can power down a SIP (which automatically deactivates any installed SPAs) and remove the SIP with the SPAs still intact.

Although graceful deactivation of a SIP is preferred using the **no power enable module** command, the Cisco 7600 series router does support removal of the SIP without deactivating it first. If you plan to remove a SIP, you can deactivate the SIP first, using the **no power enable module** global configuration command. When you deactivate a SIP using this command, it automatically deactivates each of the SPAs that are installed in that SIP. Therefore, it is not necessary to deactivate each of the SPAs prior to deactivating the SIP.

Either a blank filler plate or a functional SPA should reside in every subslot of a SIP during normal operation.

For more information about the recommended procedures for physical removal of the SIP, refer to the Cisco 7600 Series Router SIP, SSC, and SPA Hardware Installation Guide.

Deactivating a SIP or SSC

To deactivate a SIP or SSC and its installed SPAs prior to removal of the SIP, use the following command in global configuration mode:

Command	Purpose
Router(config)# **no power enable module** *slot*	Shuts down any installed interfaces, and deactivates the SIP in the specified slot, where: • *slot*—Specifies the chassis slot number where the SIP is installed.

For more information about chassis slot numbering, refer to the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section in this guide.

Reactivating a SIP or SSC

Once you deactivate a SIP or SSC, whether or not you have performed an OIR, you must use the **power enable module** global configuration command to reactivate the SIP.

If you did not issue a command to deactivate the SPAs installed in a SIP, but you did deactivate the SIP using the **no power enable module** command, then you do not need to reactivate the SPAs after an OIR of the SIP. The installed SPAs automatically reactivate upon reactivation of the SIP in the router.

For example, consider the case where you remove a SIP from the router to replace it with another SIP. You reinstall the same SPAs into the new SIP. When you enter the **power enable module** command on the router, the SPAs will automatically reactivate with the new SIP.

To activate a SIP and its installed SPAs after the SIP has been deactivated, use the following command in global configuration mode:

Command	Purpose
Router(config)# **power enable module** *slot*	Activates the SIP in the specified slot and its installed SPAs, where: • *slot*—Specifies the chassis slot number where the SIP is installed.

For more information about chassis slot numbering, refer to the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section in this guide.

Verifying Deactivation and Activation of a SIP or SSC

To verify the deactivation of a SIP or SSC, enter the **show module** command in privileged EXEC configuration mode. Observe the Status field associated with the SIP that you want to verify.

The following example shows that the Cisco 7600 SIP-400 located in slot 13 is deactivated. This is indicated by its "PwrDown" status.

```
Router# show module 13
Mod Ports Card Type                              Model              Serial No.
--- ----- -------------------------------------- ------------------ -----------

 13    0  4-subslot SPA Interface Processor-400  7600-SIP-400       JAB0851042X

Mod MAC addresses                       Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
 13  00e0.aabb.cc00 to 00e0.aabb.cc3f   0.525 12.2(PP_SPL_ 12.2(PP_SPL_ Ok

Mod Online Diag Status
--- -------------------
 13 PwrDown
```

To verify activation and proper operation of a SIP, enter the **show module** command and observe "Ok" in the Status field as shown in the following example:

```
Router# show module 2
Mod Ports Card Type                              Model              Serial No.
--- ----- -------------------------------------- ------------------ -----------
  2    0 4-subslot SPA Interface Processor-200   7600-SIP-200       JAB074905S1

Mod MAC addresses                       Hw    Fw           Sw           Status
--- ---------------------------------- ------ ------------ ------------ -------
  2  0000.0000.0000 to 0000.0000.003f   0.232 12.2(2004082 12.2(2004082 Ok

Mod Online Diag Status
--- -------------------
  2 Pass
```

Preparing for Online Removal of a SPA

The Cisco 7600 series router supports OIR of a SPA independently of removing the SIP or SSC. This means that a SIP can remain installed in the router with one SPA remaining active, while you remove another SPA from one of the SIP subslots. If you are not planning to immediately replace a SPA into the SIP, then be sure to install a blank filler plate in the subslot. The SIP should always be fully installed with either functional SPAs or blank filler plates.

The interface configuration is retained (recalled) if a SIP or SPA is removed and then replaced with one of the same type. This is not the case if you replace a Cisco 7600 SIP-200 with a Cisco 7600 SIP-400 or vice versa.

If you are planning to remove a SIP along with its SPAs, then you do not need to follow the instructions in this section. To remove a SIP, see the "Preparing for Online Removal of a SIP or SSC" section on page 5-4.

Deactivating a SPA

Although graceful deactivation of a SPA is preferred using the **hw-module subslot shutdown** command, the Cisco 7600 series router does support removal of the SPA without deactivating it first. Before deactivating a SPA, ensure that the SIP is seated securely into the slot before pulling out the SPA itself.

Note If you are preparing for an OIR of a SPA, it is not necessary to independently shut down each of the interfaces prior to deactivation of the SPA. The **hw-module subslot shutdown** command automatically stops traffic on the interfaces and deactivates them along with the SPA in preparation for OIR. In similar fashion, you do not need to independently restart any interfaces on a SPA after OIR of a SPA or SIP.

To deactivate a SPA and all of its interfaces prior to removal of the SPA, use the following command in global configuration mode:

Command	Purpose
Router(config)# **hw-module subslot** *slot/subslot* **shutdown** [**powered** | **unpowered**]	Deactivates the SPA in the specified slot and subslot of the SIP, where: • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies subslot number on a SIP where a SPA is installed. • **powered**—(Optional) Shuts down the SPA and all of its interfaces, and leaves them in an administratively down state with power enabled. This is the default state. • **unpowered**—(Optional) Shuts down the SPA and all of its interfaces, and leaves them in an administratively down state without power.

For more information about chassis slot and SIP subslot numbering, refer to the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section in this guide.

Reactivating a SPA

Note You do not need to reactivate a SPA after an OIR of either the SIP or a SPA if you did not deactivate the SPA prior to removal. If the router is running, then the SPAs automatically start upon insertion into the SIP or with insertion of a SIP into the router.

If you deactivate a SPA using the **hw-module subslot shutdown** global configuration command and need to reactivate it without performing an OIR, you need to use the **no hw-module subslot shutdown** global configuration command to reactivate the SPA and its interfaces.

To activate a SPA and its interfaces after the SPA has been deactivated, use the following command in global configuration mode:

Command	Purpose
Router(config)# **no hw-module subslot** *slot/subslot* **shutdown**	Activates the SPA and its interfaces in the specified slot and subslot of the SIP, where: • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies subslot number on a SIP where a SPA is installed.

Verifying Deactivation and Activation of a SPA

When you deactivate a SPA, the corresponding interfaces are also deactivated. This means that these interfaces will no longer appear in the output of the **show interface** command.

To verify the deactivation of a SPA, enter the **show hw-module subslot all oir** command in privileged EXEC configuration mode. Observe the Operational Status field associated with the SPA that you want to verify.

In the following example, the SPA located in subslot 1 of the SIP in slot 2 of the router is administratively down from the **hw-module subslot shutdown** command:

```
Router# show hw-module subslot all oir
Module         Model              Operational Status
-------------- ------------------ -------------------------
subslot 2/0    SPA-4XOC3-POS      ok
subslot 2/1    SPA-4XOC3-ATM      admin down
```

To verify activation and proper operation of a SPA, enter the **show hw-module subslot all oir** command and observe "ok" in the Operational Status field as shown in the following example:

```
Router# show hw-module subslot all oir
Module         Model              Operational Status
-------------- ------------------ -------------------------
subslot 2/0    SPA-4XOC3-POS      ok
subslot 2/1    SPA-4XOC3-ATM      ok
```

Deactivation and Activation Configuration Examples

This section provides the following examples of deactivating and activating SIPs and SPAs:

- Deactivation of a SIP Configuration Example, page 5-8
- Activation of a SIP Configuration Example, page 5-8
- Deactivation of a SPA Configuration Example, page 5-8
- Activation of a SPA Configuration Example, page 5-8

Deactivation of a SIP Configuration Example

Deactivate a SIP when you want to perform OIR of the SIP. The following example deactivates the SIP that is installed in slot 5 of the router, its SPAs, and all of the interfaces. The corresponding console messages are shown:

```
Router# configure terminal
Router(config)# no power enable module 5
1w4d: %OIR-6-REMCARD: Card removed from slot 5, interfaces disabled
1w4d: %C6KPWR-SP-4-DISABLED: power to module in slot 5 set off (admin request)
```

Activation of a SIP Configuration Example

Activate a SIP if you have previously deactivated it. If you did not deactivate the SPAs, the SPAs automatically reactivate with reactivation of the SIP.

The following example activates the SIP that is installed in slot 5 of the router, its SPA, and all of the interfaces (as long as the **hw-module subslot shutdown** command was not issued to also deactivate the SPA):

```
Router# configure terminal
Router(config)# power enable module 5
```

Notice that there are no corresponding console messages shown with activation. If you re-enter the **power enable module** command, a message is displayed indicating that the module is already enabled:

```
Router(config)# power enable module 5
% module is already enabled
```

Deactivation of a SPA Configuration Example

Deactivate a SPA when you want to perform OIR of that SPA. The following example deactivates the SPA (and its interfaces) that is installed in subslot 0 of the SIP located in slot 2 of the router and removes power to the SPA. Notice that no corresponding console messages are shown:

```
Router# configure terminal
Router(config)# hw-module subslot 2/0 shutdown unpowered
```

Activation of a SPA Configuration Example

Activate a SPA if you have previously deactivated it. If you have not deactivated a SPA and its interfaces during OIR of a SIP, then the SPA is automatically reactivated upon reactivation of the SIP.

The following example activates the SPA that is installed in slot 2 of the router and all of its interfaces.

```
Router# configure terminal
Router(config)# no hw-module subslot 2/0 shutdown
Router#
```

PART 3

ATM Shared Port Adapters

CHAPTER 6

Overview of the ATM SPAs

This chapter provides an overview of the release history, features, and MIB support for the 1-Port OC-48c/STM-16 ATM SPA, 1-Port OC-12c/STM-4 ATM SPA, and the 2-Port and 4-Port OC-3c/STM-1 ATM SPA. This chapter includes the following sections:

Release History

Release	Modification
12.2(33)SRC	• Support for Phase 2 Access Circuit Redundancy with Local Switching
12.2(33)SRA	• Some restrictions for QoS and MLPPP bundles were added. • Support for the following features was introduced for ATM SPAs on the Cisco 7600 SIP-200: – AToM VP Mode Cell Relay – MPLS over RBE – Multi-VC to VLAN scalability – QoS support on bridging features • Support for the following features was introduced for ATM SPAs on the Cisco 7600 SIP-400: – AToM VP Mode Cell Relay – Multi-VC to VLAN scalability – Multi-VLAN to VC – QoS support on bridging features
12.2(18)SXE	• Support was introduced for the 2-Port and 4-Port OC-3c/STM-1 ATM SPAs on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400 SPA interface processors (SIPs) on the Cisco 7600 series router and Catalyst 6500 series switch. • Support was introduced for the 1-Port OC-12c/STM-4 ATM SPA on the Cisco 7600 SIP-400 on the Cisco 7600 series router and Catalyst 6500 series switch.
12.2(18)SXF	• Support was introduced for the 1-Port OC-48c/STM-16 ATM SPA on the Cisco 7600 SIP-400 on the Cisco 7600 series router and Catalyst 6500 series switch.
12.2(18)SXF2	• Support for the "Enhancements to RFC 1483 Spanning Tree Interoperability" feature was added for ATM SPAs on the Cisco 7600 series router and Catalyst 6500 series switch. • Documentation of a workaround for ATM SPA configuration on the Cisco 7600 SIP-200 has been added in Chapter 7, "Configuring the ATM SPAs" to address a Routed Bridge Encapsulation (RBE) limitation where only one remote MAC address is supported.

Overview

The ATM SPAs are single-width, double-height, cross-platform Optical Carrier (OC) ATM adapter cards that provide OC-3c/STM-1c (155.52 Mbps), OC-12c/STM-4c (622.080 Mbps), or OC-48/STM-16 (2488 Mbps) connectivity and can be used in a Cisco 7600 series router. The ATM SPAs come in the following models:

- 2-Port and 4-Port OC-3c/STM-1 ATM SPA (SPA-2XOC3-ATM=, SPA-4XOC3-ATM=)
- 1-Port OC-12c/STM-4 POS SPA (SPA-1XOC12-ATM=)

- 1-Port OC-48c/STM-16 ATM SPA (SPA-1XOC48-ATM=)

The OC-3c ATM SPAs must be installed in a Cisco 7600 SIP-200 or Cisco 7600 SIP-400 SPA interface processor (SIP) before they can be used in the Cisco 7600 series router. The 1-Port OC-12c/STM-4 ATM SPA and 1-Port OC-48c/STM-16 ATM SPA must be installed in a Cisco 7600 SIP-400 before they can be used in the Cisco 7600 series router.

You can install the SPA in the SIP before or after you insert the SIP into the router chassis. This allows you to perform online insertion and removal (OIR) operations either by removing individual SPAs from the SIP, or by removing the entire SIP (and its contained SPAs) from the router chassis.

The ATM SPAs provide cost-effective wide-area network (WAN) connectivity for service providers across their existing ATM networks. Using a highly modular approach, the SPA and SIP form factors maximize the flexibility of an existing Cisco 7600 series router, allowing service providers to mix and match SPAs to more easily meet evolving port-density and networking media needs.

The ATM SPAs also use small form-factor pluggable (SFP) optical transceivers, giving service providers port-level flexibility for different types of optical media (such as single mode and multimode). Changing the type of optical network involves simply replacing the transceiver, not the SPAs or SIP.

Note A maximum of two ATM SPAs can be installed in each SIP, and these SPAs can be different models (such as a 2-Port OC-3c/STM-1 ATM SPA and a 1-Port OC-12c/STM-4 ATM SPA). You can also mix SPAs of different types, such as ATM and POS, in a SIP, depending on the space requirements of the SIPs. An exception is that only one 1-Port OC-48c/STM-16 ATM SPA can be installed in a SIP; the other slot should be left empty.

See the following sections for more information about the ATM SPAs:

- ATM Overview, page 6-3
- PVC and SVC Encapsulations, page 6-4
- PVC and SVC Service Classes, page 6-4
- Advanced Quality of Service, page 6-5

ATM Overview

Asynchronous Transfer Mode (ATM) uses cell-switching and multiplexing technology that combines the benefits of circuit switching (constant transmission delay and guaranteed capacity) with those of packet switching (flexibility and efficiency for intermittent traffic). ATM transmits small cells (53 bytes) with minimal overhead (5 bytes of header and checksum, with 48 bytes for data payload), allowing for very quick switching times between the input and output interfaces on a router.

ATM is a connection-oriented environment, in which each ATM endpoint (or node) must establish a separate connection to the specific endpoints in the ATM network with which it wants to exchange traffic. This connection (or channel) between the two endpoints is called a *virtual circuit* (VC).

Each VC is uniquely identified by the combination of a virtual path identifier (VPI) and a virtual channel identifier (VCI). The VC is treated as a point-to-point mechanism to another router or host and is capable of supporting bidirectional traffic.

In an ATM network, a VC can be either a permanent virtual circuit (PVC) or a switched virtual circuit (SVC). A network operator must manually configure a PVC, which remains in force until it is manually torn down. An SVC is set up and torn down using an ATM signaling mechanism. On the ATM SPAs, this signaling is based on the ATM Forum User-Network Interface (UNI) specification V3.x and V4.0.

PVC and SVC Encapsulations

PVCs and SVCs are configured with an ATM encapsulation type that is based upon the ATM Adaptation Layer (AAL). The following types are supported:

- AAL5CISCOPPP—AAL5 Cisco PPP encapsulation, which is Cisco's proprietary PPP over ATM encapsulation.
- AAL5MUX—ATM Adaptation Layer 5 MUX encapsulation, also known as null encapsulation, that supports a single protocol (IP or IPX).
- AAL5NLPID—(Supported on ATM SPAs in a Cisco 7600 SIP-200 only) AAL5 Network Layer Protocol Identification (NLPID) encapsulation, which allows ATM interfaces to interoperate with High-Speed Serial Interfaces (HSSIs) that are using an ATM data service unit (ADSU) and running ATM-Data Exchange Interface (DXI).
- AAL5SNAP—AAL5 Logical Link Control/Subnetwork Access Protocol (LLC/SNAP) encapsulation, which supports Inverse ARP and incorporates the LLC/SNAP that precedes the protocol datagram. This allows the use of multiple protocols over the same VC, and is particularly well–suited for encapsulating IP packets.

Note The 1-Port OC-48c/STM-16 ATM SPA supports only AAL5MUX and AAL5SNAP encapsulations.

PVC and SVC Service Classes

ATM was designed with built-in quality of service capabilities to allow it to efficiently multiplex different types of traffic over the same links. To accomplish this, each PVC or SVC is configured with a service class that defines the traffic parameters, such as maximum cell rate or burst rate, for the circuit. The following service classes are available in ATM networks:

- Constant Bit Rate (CBR)—The ATM router transmits ATM cells in a continuous bit-stream that is suitable for real-time traffic, such as voice and video. CBR is typically used for VCs that need a static amount of bandwidth (constant bit rate or average cell rate) that is continuously available for the duration of the active connection. The ATM router guarantees that a VC with a CBR service class can send cells at the peak cell rate (PCR) at any time, but the VC is also free to use only part of the allocated bandwidth, or none of the bandwidth, as well.
- Unspecified Bit Rate (UBR)—The ATM router does not make any quality of service (QoS) commitment at all to the PVC or SVC, but instead uses a best-effort attempt to send the traffic transmitted by the PVC or SVC. UBR typically is the default configuration and is used for non-critical Internet connectivity, including e–mail, file transfers, web browsing, and so forth. The ATM router enforces a maximum peak cell rate (PCR) for the VC, to prevent the VC from using all the bandwidth that is available on the line.
- Unspecified Bit Rate Plus (UBR+)—UBR+ is a special ATM service class developed by Cisco Systems. UBR+ uses MCR (Minimum Cell Rate) along with PCR (Peak Cell Rate). In UBR+, the MCR is a "soft guarantee" of minimum bandwidth. A router signals the MCR value at call setup time when a switched VC is created. The ATM router is then responsible for the guarantee of the bandwidth specified in the MCR parameter. A UBR+ VC is a UBR VC for which the MCR is signaled by the router and guaranteed by the ATM router. Therefore, UBR+ affects connection admission control and resource allocation on ATM routers. The UBR+ service class is supported only on SVCs for an ATM SPA. It is not supported on PVCs for an ATM SPA.

Note UBR+ is not supported on the 1-Port OC-48c/STM-16 ATM SPA.

- Variable Bit Rate–Non-Real Time (VBR–nrt)—The ATM router attempts to guarantee a minimum burst size (MBS) and sustainable cell rate (SCR) for non-real-time traffic that is bursty in nature, such as database queries or aggregation of large volumes of traffic from many different sources. The ATM router also enforces a maximum peak cell rate (PCR) for the VC, to prevent the VC from using all of the bandwidth that is available on the line.
- Variable Bit Rate–Real Time (VBR–rt)—The ATM router guarantees a maximum burst size (MBS) and sustainable cell rate (SCR) for real-time traffic that is bursty in nature, such as voice, video conferencing, and multiplayer gaming. VBR-rt traffic has a higher priority than VBR-nrt traffic, allowing the real-time traffic to preempt the non-real-time traffic, if necessary. The ATM router also enforces a maximum peak cell rate (PCR) for the VC, to prevent the VC from using all the bandwidth that is available on the line.

Note The ATM SPAs do not support the Available Bit Rate (ABR) service class, which uses a minimum cell rate (MCR).

Advanced Quality of Service

In addition to the integrated QoS capabilities that are provided by the standard ATM service classes, the ATM SPA cards support a number of advanced QoS features. These features include the following:

- Per-VC and Per-VP Traffic Shaping—Enables service providers to control the bandwidth provided at the VC or VP level. You cannot shape a VC that is part of a shaped VP. You can however enable both VC and VP shaping simultaneously (as long as shaped VCs use a different VPI value than the shaped VP).
- Layer 3 (IP) QoS at the Per-VC Level—Allows marking and classifying traffic at the IP layer, for each VC, enabling service providers to control the individual traffic flows for a customer, so as to meet the customer's particular QoS needs. The IP QoS can use the IP type of service (ToS) bits, the RFC 2475 Differentiated Services Code Point (DSCP) bits, and the MPLS EXP bits. WRED, LLQ, CBWFQ, policing, classification, and marking are supported.
- Multiprotocol Label Switching (MPLS)—Allows service providers to provide cost-effective virtual private networks (VPNs) to their customers, while simplifying load balancing and QoS management, without incurring the overhead of extensive Layer 3 routing.
- IP to ATM Mapping—Creates a mapping between the Cell Loss Priority (CLP) bit in ATM cell headers and the IP precedence or IP Differentiated Services Code Point (DSCP) bits.
- VC Bundling—Selects the output VC on the basis of the IP Class of Service (CoS) bits. (Supported only when using the Cisco 7600 SIP-200 and not the Cisco 7600 SIP-400.)

Note Additional QoS features are expected to be added with each Cisco IOS software release. Please see the release notes for each release for additional features that might be supported and for the restrictions that might affect existing features.

Supported Features

This section provides a list of some of the primary features supported by the ATM hardware and software:

SIP-Dependent Features

Most features for the ATM SPAs are supported on both the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, but some features are supported only on a particular model of SIP. Table 6-1 lists the features that are supported on only one model of SIP. Any supported features for the ATM SPAs that are not listed in this table are supported on both SIPs.

Table 6-1 SIP-Dependent Feature Support

Feature	Supported on Cisco 7600 SIP-200	Supported on Cisco 7600 SIP-400
AAL5NLPID encapsulation and Routed-NLPID-PDUs	Yes	No
ATM VC Access Trunk Emulation (multi-VLAN to VC)	Yes	Yes
Bridging of Routed Encapsulations (BRE)	Yes	No
Frame Relay to ATM (FR-ATM) internetworking	No	No
Network-Based Application Recognition (NBAR)	Yes	No
RFC-1483 ATM Half-Bridging and Routed Bridged Encapsulation (RBE)	Yes	No
VC Bundling (Selects the output VC on the basis of the IP CoS bits)	Yes	No
RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*, Multipoint Bridging (MPB) (also known as multi-VC to VLAN) on the 2-Port and 4-Port OC-3c/STM-1c ATM SPA	Yes	Yes
Aggregate WRED	Yes	Yes

Basic Features

- Bellcore GR-253-CORE SONET/SDH compliance (ITU-T G.707, G.783, G.957, G.958)
- Interface-compatible with other Cisco ATM adapters

> **Note** The ATM SPA is functionally similar to other ATM port adapters on the Cisco 7600 series router, but because it is a different card type, the configuration for the slot is lost when you replace an existing ATM port adapter with an ATM SPA in a SIP.

- Supports both permanent virtual circuits (PVCs) and switched virtual circuits (SVCs)
- An absolute maximum of 16,384 (16K) configured VCs per ATM SPA (4,096 [4K] per interface) with the following recommended limitations:
  - On a Cisco 7600 SIP-400, 8000 PVCs are supported on multipoint subinterfaces. The limit of 16,384 PVCs only applies to the Cisco 7600 SIP-200.
  - A recommended maximum number of 2,048 PVCs on all point-to-point subinterfaces for all ATM SPAs in a SIP.
  - A recommended maximum number of 16,380 PVCs on all multipoint subinterfaces for all ATM SPAs in a SIP, and a recommended maximum number of 200 PVCs per each individual multipoint subinterface.
  - A recommended maximum number of 400 SVCs for all ATM SPAs in a SIP.
  - A recommended maximum number of 1,024 PVCs or 400 SVCs using service policies for all ATM SPAs in a SIP.
- Up to 4,096 simultaneous segmentations and reassemblies (SARs) per interface
- Supports a maximum number of 200 PVCs or SVCs using Link Fragmentation and Interleaving (LFI) for all ATM SPAs (or other ATM modules) in a Cisco 7600 series router
- A maximum number of 1000 PVCs or 400 SVCs configured with Modular QoS CLI (MQC) policy maps
- Up to 1,000 maximum virtual templates per router
- ATM adaptation layer 5 (AAL5) for data traffic
- Hardware switching of multicast packets for point-to-point subinterfaces
- SONET/SDH (software selectable) optical fiber (2-Port and 4-Port OC-3c/STM-1 ATM SPA, 1-Port OC-48c/STM-16 ATM SPA, or 1-Port OC-12c/STM-4 ATM SPA), depending on the model of ATM SPA
- Uses small form-factor pluggable (SFP) optical transceivers, allowing the same ATM SPA hardware to support multimode (MM), single-mode intermediate (SMI), or single-mode long (SML) reach, depending on the capabilities of the SPA
- ATM section, line, and path alarm indication signal (AIS) cells, including support for F4 and F5 flows, loopback, and remote defect indication (RDI)
- Operation, Administration, and Maintenance (OAM) cells
- Online insertion and removal (OIR) of individual ATM SPAs from the SIP, as well as OIR of the SIPs with ATM SPAs installed

SONET/SDH Error, Alarm, and Performance Monitoring

- Fiber removed and reinserted
- Signal failure bit error rate (SF-BER)
- Signal degrade bit error rate (SD-BER)
- Signal label payload construction (C2)
- Path trace byte (J1)
- Section Diagnostics:
  - Loss of signal (SLOS)
  - Loss of frame (SLOF)
  - Error counts for B1
  - Threshold crossing alarms (TCA) for B1 (B1-TCA)
- Line Diagnostics:
  - Line alarm indication signal (LAIS)
  - Line remote defect indication (LRDI)
  - Line remote error indication (LREI)
  - Error counts for B2
  - Threshold crossing alarms for B2 (B2-TCA)
- Path Diagnostics:
  - Path alarm indication signal (PAIS)
  - Path remote defect indication (PRDI)
  - Path remote error indication (PREI)
  - Error counts for B3
  - Threshold crossing alarms for B3 (B3-TCA)
  - Loss of pointer (PLOP)
  - New pointer events (NEWPTR)
  - Positive stuffing event (PSE)
  - Negative stuffing event (NSE)
- The following loopback tests are supported:
  - Network (line) loopback
  - Internal (diagnostic) loopback
- Supported SONET/SDH synchronization:
  - Local (internal) timing (for inter-router connections over dark fiber or wavelength division multiplexing [WDM] equipment)
  - Loop (line) timing (for connecting to SONET/SDH equipment)
  - +/– 4.6 ppm clock accuracy over full operating temperature

Layer 2 Features

- Supports the following encapsulation types:
  - AAL5SNAP (LLC/SNAP)
  - LLC encapsulated bridged protocol
  - AAL5MUX (VC multiplexing)
  - AAL5NLPID and Routed-NLPID-PDUs (ATM SPAs in a Cisco 7600 SIP-200 only)
  - AAL5CISCOPPP
- Supports the following ATM traffic classes and per-VC traffic shaping modes:
  - Constant bit rate (CBR) with peak rate
  - Unspecified bit rate (UBR) with peak cell rate (PCR)
  - Non-real-time variable bit rate (VBR-nrt)
  - Variable bit rate real-time (VBR-rt)
  - Unspecified bit rate plus (UBR+) on SVCs

Note ATM shaping is supported, but class queue-based shaping is not.

- ATM point-to-point and multipoint connections
- Explicit Forward Congestion Indication (EFCI) bit in the ATM cell header
- Frame Relay to ATM (FR-ATM) internetworking (ATM SPAs in a Cisco 7600 SIP-200 only)
- Integrated Local Management Interface (ILMI) operation, including keepalive, PVC discovery, and address registration and deregistration
- Link Fragmentation and Interleaving (LFI) performed in hardware
- VC–to–VC local switching and cell relay
- VP–to–VP local switching and cell relay
- AToM VP Mode Cell Relay support
- RFC 1755, *ATM Signaling Support for IP over ATM*
- ATM User-Network Interface (UNI) signalling V3.0, V3.1, and V4.0 only
- RFC 2225, *Classical IP and ARP over ATM* (obsoletes RFC 1577)
- Unspecified bit rate plus (UBR+) traffic service class on SVCs

Layer 3 Features

- ATM VC Access Trunk Emulation (multi-VLAN to VC) (ATM SPAs in a Cisco 7600 SIP-200 only)
- ATM over MPLS (AToM) in AAL5 mode (except for AToM cell packing)
- ATM over MPLS (AToM) in AAL5/AAL0 VC mode
- Bridging of Routed Encapsulations (BRE) (ATM SPAs in a Cisco 7600 SIP-200 only)
- Distributed Link Fragmentation and Interleaving (dLFI) for ATM (dLFI packet counters are supported, but dLFI byte counters are not supported)

- LFI with dCRTP
- Network-Based Application Recognition (NBAR) (ATM SPAs in a Cisco 7600 SIP-200 only)
- No limitation on the maximum number of VCs per VPI, up to the maximum number of 4,096 total VCs per interface (so there is no need to configure this limit using the **atm vc-per-vp** command, which is required on other ATM SPAs)
- OAM flow connectivity using OAM ping for segment or end-to-end loopback
- PVC multicast (Protocol Independent Multicast [PIM] dense and sparse modes)
- Quality of Service (QoS):
  - Policing
  - IP-to-ATM class of service (IP precedence and DSCP)
  - Per-VC class-based weighted fair queueing (CBWFQ)
  - Per-VC Layer 3 queueing
  - VC Bundling (Cisco 7600 SIP-200 only)
  - Weighted Random Early Detection (WRED)
  - Aggregate WRED
- RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*:
  - Routed Bridge Encapsulation (RBE) (ATM SPAs in a Cisco 7600 SIP-200 only)
  - Half-bridging (ATM SPAs in a Cisco 7600 SIP-200 only)
  - PVC bridging (full-bridging) on Cisco 7600 SIP-200 and Cisco 7600 SIP-400
- Supports oversubscription by default
- Routing protocols:
  - Border Gateway Protocol (BGP)
  - Enhanced Interior Gateway Routing Protocol (EIGRP)
  - Interior Gateway Routing Protocol (IGRP)
  - Integrated Intermediate System-to-Intermediate System (IS-IS)
  - Open Shortest Path First (OSPF)
  - Routing Information Protocol version 1 and version 2 (RIPv1 and RIPv2)

High Availability Features

- 1+1 Automatic Protection Switching (APS) redundancy (PVC circuits only)
- Route Processor Redundancy (RPR)
- RPR Plus (RPR+)
- OSPF Nonstop Forwarding (NSF)
- Stateful Switchover (SSO)

Enhancements to RFC 1483 Spanning Tree Interoperability

This section describes an interoperability feature for the various spanning tree implementations across 1483 Bridge Mode ATM PVCs. Historically, vendors have not implemented spanning tree across RFC 1483 encapsulation consistently; furthermore, some Cisco IOS releases may not support the full range of spanning tree options. This feature attempts to smooth some of the practical challenges of interworking common variations of spanning tree over RFC 1483 Bridge Mode encapsulation.

Note This feature set is only supported on RFC 1483 Bridge Mode ATM permanent virtual circuits (PVCs).

Some basic terms include the following:

- *IEEE 802.1D* is a standard for interconnecting LANs through media access control (MAC) bridges. IEEE 802.1D uses the Spanning Tree Protocol to eliminate loops in the bridge topology, which cause broadcast storms.
- *Spanning Tree Protocol (STP)* as defined in IEEE 802.1D is a link-management protocol that provides path redundancy while preventing undesirable loops in the network. An IEEE 802.1D spanning tree makes it possible to have one spanning tree instance for the whole switch, regardless of the number of VLANs configured on the switch.
- *Bridge Protocol Data Unit (BPDU)* is the generic name for the frame used by the various spanning tree implementations. The Spanning Tree Protocol uses the BPDU information to elect the root switch and root port for the switched network, as well as the root port and designated port for each switched segment.
- *Per VLAN Spanning Tree (PVST)* is a Cisco proprietary protocol that allows a Cisco device to support multiple spanning tree topologies on a per-VLAN basis. PVST uses the BPDUs defined in IEEE 802.1D (see Figure 6-2 on page 6-13), but instead of one STP instance per switch, there is one STP instance per VLAN.
- *PVST+* is a Cisco proprietary protocol that creates one STP instance per VLAN (as in PVST). However, PVST+ enhances PVST and uses Cisco proprietary BPDUs with a special 802.2 Subnetwork Access Protocol (SNAP) Organizational Unique Identifier (OUI)[1] (see Figure 6-2 on page 6-13) instead of the standard IEEE 802.1D frame format used by PVST. PVST+ BPDUs are also known as Simple Symmetric Transmission Protocol (SSTP) BPDUs.

Note RFC 1483 is referenced throughout this section, although it has been superseded by RFC 2684.

1. The Organizational Unique Identifier (OUI) portion of the MAC address often identifies the vendor of the upper layer protocol or the manufacturer of the Ethernet adapter. The OUI value of 00-00-0C identifies Cisco Systems as the manufacturer of the Ethernet adapter.

Supported Supervisor Engines and Line Cards

The Cisco 7600 series routers support PVST to PVST+ BPDU interoperability with the Cisco 7600 SIP-200.

The Interoperability Problem

The current interoperability problem can be summarized as follows:

- When transmitting STP BPDUs, many vendors' implementations of ATM-to-Ethernet bridging are not fully compliant with the specifications of RFC 1483, Appendix B. The most common variation of the standard is to use an ATM Common Part Convergence Sublayer (CPCS) SNAP protocol data unit (PDU) with OUI: 00-80-C2 and PID: 00-07. Appendix B reserved this OUI/PID combination for generic Ethernet frames without BPDUs. Appendix B specifies OUI: 00-80-C2 and protocol identifier (PID): 00-0E for frames with BPDU contents.
- There are several varieties of the Spanning Tree Protocol used by Cisco products on ATM interfaces. The Catalyst 5000 series supports only PVST on ATM interfaces. The Cisco 7600 series router and Catalyst 6500 series switches support only PVST+ on ATM interfaces. Most other Cisco routers implement classic IEEE 802.1D on ATM interfaces.

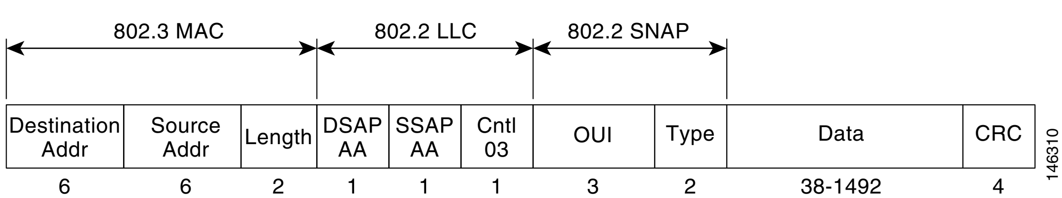
  When the Cisco 7600 series router and the Catalyst 6500 series switch first implemented RFC 1483 Bridging (on Cisco IOS Release 12.1E) on the Cisco 7600 FlexWAN module, the platform used OUI: 00-80-C2 and PID: 00-0E to maximize interoperability with all other Cisco IOS products.

  However, there are so many implementations that do not send PVST or IEEE 802.1D BPDUs with PID: 00-0E that the Cisco 7600 series routers and the Catalyst 6500 series switches reverted to the more common implementation of RFC 1483 (with PID: 00-07) in Cisco IOS Release 12.2SX. This spanning tree interoperability feature provides the option of encapsulating BPDUs across RFC 1483 with either PID: 00-07 or PID: 00-0E.

BPDU Packet Formats

The various BPDU packet formats are described in this section. Figure 6-1 shows the generic IEEE 802.2/802.3 frame format, which is used by PVST+, but is not used by PVST.

Figure 6-1 IEEE 802.2/802.3 SNAP Encapsulation Frame Format



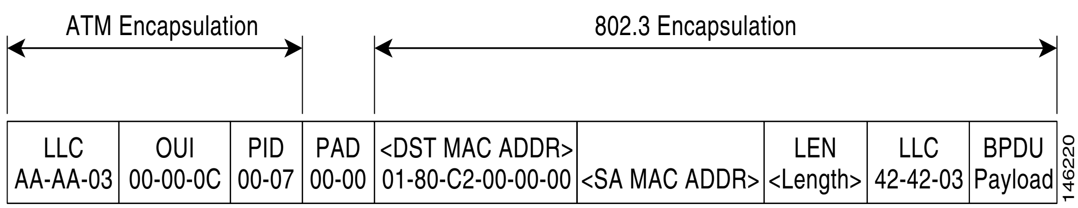
In an Ethernet SNAP frame, the *SSAP* and *DSAP* fields are always set to AA. These codes identify it as a SNAP frame. The Control field always has a value of 03, which specifies connectionless logical link control (LLC) services.

The Type field identifies the upper layer protocol to which data should be passed. For example, a *Type* field of hex 0800 represents IP, while a value of 8137 indicates that data is meant for IPX.

Catalyst 5000 PVST BPDU Packet Format

The Catalyst 5000 series switches send and receive BPDUs in PVST format on ATM interfaces (see Figure 6-2).

Figure 6-2 BPDU PVST Frame Format Used by the Catalyst 5000 Switch



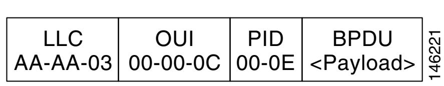
- BPDUs sent by the Catalyst 5000 series switch use a PID of 0x00-07, which does not comply with RFC 1483. The Cisco 7600 series router also has the ability to send BPDUs in this data format.
- The PAD portion of the ATM encapsulation varies from 0 to 47 bytes in length to ensure complete ATM cell payloads.
- By using the **bridge-domain** command's **ignore-bpdu-pid** optional keyword, the Catalyst 5000 series switch sends this frame by default.
- The Catalyst 5000 series switch cannot accept the PVST+ BPDUs and blocks the ATM port, giving the following error messages:

```
%SPANTREE-2-RX_1QNON1QTRUNK: Rcved 1Q-BPDU on non-1Q-trun port 6/1 vlan 10
%SPANTREE-2-RX_BLKPORTPVID: Block 6/1 on rcving vlan 10 for inc peer vlan 0
```

Cisco 7200 and Cisco 7500 Series Routers IEEE 802.1D BPDU Frame Format

Figure 6-3 shows the Cisco 7200 and Cisco 7500 series routers IEEE 802.1D BPDU frame format.

Figure 6-3 Frame Format for the Cisco 7200 and Cisco 7500 Series Routers IEEE 802.1D BPDU



Cisco 7600 Router PVST+ BPDU Frame Format

The Cisco 7600 series router PVST+ BPDU packet format is shown in Figure 6-4. These BPDUs are not IEEE 802.1D BPDUs, but Cisco proprietary SSTP BPDUs.

Figure 6-4 Cisco 7600 Router PVST+ BPDU Frame Format (1483 Bridge Mode)

ATM Encapsulation (LLC, OUI, PID)

LLC	OUI	PID	PAD	DA (SSTP DA MAC)	SA	LEN	LLC	OUI	Type (SSTP)	BPDU
AA-AA-03	00-80-C2	00-07	00-00	01-00-0C-CC-CC-CD	<SA MAC>	<Length>	AA-AA-03	00-00-0C	01-0B	<Payload>

146222

Cisco L2PT BPDU Frame Format

Figure 6-5 shows the Cisco Layer 2 Protocol Tunneling (L2PT) BPDU SNAP frame format.

Figure 6-5 L2PT BPDU SNAP Frame Format

DA (L2PTDA MAC)	SA	LEN	LLC	OUI	Type (SSTP)	BPDU
01-00-0C-CD-CD-D0	<SA MAC>	<Length>	AA-AA-03	00-00-0C	01-0B	<Payload>

146223

Unsupported Features

- The following High Availability features are not supported:
  - APS N+1 redundancy is not supported.
  - APS redundancy is not supported on SVCs.
  - APS reflector mode (**aps reflector** interface configuration command) is not supported.
- The **atm bridge-enable** command, which was used in previous releases on other ATM interfaces to enable multipoint bridging on PVCs, is not supported on ATM SPA interfaces. Instead, use the **bridge** option with the **encapsulation** command to enable RFC 1483 half-bridging on PVCs. See the "Configuring ATM Routed Bridge Encapsulation" section on page 7-21.
- AToM cell packing is not supported.
- PVC autoprovisioning (**create on-demand** VC class configuration command) is not supported.
- Creating SVCs with UNI signalling version 4.1 is not supported (UNI signalling v 3.0, v 3.1, and v 4.0 are supported).
- Enhanced Remote Defect Indication–Path (ERDI-P) is not supported.
- Fast Re-Route (FRR) over ATM is not supported.
- LAN Emulation (LANE) is not supported.
- Multicast SVCs are not supported.
- Available Bit Rate (ABR) traffic service class is not supported.
- Unspecified bit rate plus (UBR+) traffic service class is not supported on PVCs.

Prerequisites

- The 2-Port and 4-Port OC-3c/STM-1 ATM SPAs must use either the Cisco 7600 SIP-200 or Cisco 7600 SIP-400.
- The 1-Port OC-12c/STM-4 ATM SPA must use the Cisco 7600 SIP-400.
- The 1-Port OC-48c/STM-16 ATM SPA must use the Cisco 7600 SIP-400.
- The Cisco 7600 SIP-200 requires a Cisco 7600 series router using a SUP-720 3B and above processor that is running Cisco IOS Release 12.2(18)SXE or later release.
- The Cisco 7600 SIP-400 requires a Cisco 7600 series router using a SUP-720 processor that is running Cisco IOS Release 12.2(18)SXE or later release.
- Before beginning to configure the ATM SPA, have the following information available:
  - Protocols you plan to route on the new interfaces.
  - IP addresses for all ports on the new interfaces, including subinterfaces.
  - Bridging encapsulations you plan to use.

Restrictions

- The 1-Port OC-48c/STM-16 ATM SPA does not support the following features: AToM, BRE, LFI, RBE, SVCs, UBR+, RFC 2225 (formerly RFC 1577), or bridging.
- The ATM SPAs in the Cisco 7600 series router do not support APS reflector and reflector channel modes. (These modes require a facing path terminating element [PTE], which is typically a Cisco ATM switch.)
- The ATM SPA is functionally similar to other ATM port adapters on the Cisco 7600 series router, such as the PA-A3, but it is a different card type, so the slot's previous configuration is lost when you replace an existing ATM port adapter with an ATM SPA.
- The following restrictions apply to the operation of QoS on the ATM SPAs:
  - The ATM SPAs do not support bandwidth-limited priority queueing, but support only strict priority policy maps (that is, the **priority** command without any parameters).
  - A maximum of one **priority** command is supported in a policy map.
  - You cannot use the **match input interface** command in policy maps and class maps that are being used for ATM SPAs.
  - Hierarchical traffic shaping (traffic shaping on both the VC and VP for a circuit) is not supported. Traffic shaping can be configured only on the VC or on the VP, but not both.
  - ATM (Layer 2) output shaping is supported, but IP (Layer 3) shaping on an output (egress) interface is not supported. In particular, this means that you cannot use any **shape** class-map configuration commands in policy maps that are being used in the output direction. This includes the **shape adaptive**, **shape average**, **shape fecn-adapt**, and **shape peak** commands.
  - The ATM SPA interfaces support a maximum of six configured precedences (using the **random-detect aggregate** command) in each class map in a policy map. The maximum number of configurable subclass groups is seven.
  - STP is not supported in ATM Multi-Vlan-to-VC mode.

- For best performance, we recommend the following maximums:
  - A maximum number of 2,048 PVCs on all point-to-point subinterfaces for all ATM SPAs in a SIP.
  - A maximum number of 16,380 PVCs on all multipoint subinterfaces for all ATM SPAs in a SIP.
  - A maximum number of 400 SVCs for all ATM SPAs in a SIP.
  - A maximum number of 1024 PVCs or SVCs s using service policies for all ATM SPAs in a router.
  - A maximum number of 200 PVCs or SVCs using Link Fragmentation and Interleaving (LFI) for all ATM SPAs in a router.
  - A maximum number of 200 PVCs on each multipoint subinterface being used on an ATM SPA.

Note These limits are flexible and depend on all factors that affect performance in the router, such as processor card, type of traffic, and so on.

- In the default configuration of the transmit path trace buffer, the ATM SPA does not support automatic updates of remote host name and IP address (as displayed by the **show controllers atm** command). This information is updated only when the interface is shut down and reactivated (using the **shutdown** and **no shutdown** commands). Information for the received path trace buffer, however, is automatically updated.
- The **show ppp multilink** command displays only the packet counters, and not byte counters, for a dLFI configuration on an ATM SPA interface.
- MLPPP is supported, but not MLPPP bundles.

Supported MIBs

The following MIBs are supported in Cisco IOS Release 12.2(18)SXE and later releases for the ATM SPAs on the Cisco 7600 series router.

Common MIBs

- ENTITY-MIB
- IF-MIB
- MIB-II
- MPLS-CEM-MIB

Cisco-Specific Common MIBs

- CISCO-ENTITY-EXT-MIB
- OLD-CISCO-CHASSIS-MIB
- CISCO-CLASS-BASED-QOS-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-SENSOR-MIB
- CISCO-MQC-MIB

Cisco-Specific MPLS MIBs

- CISCO-IETF-PW-MIB
- CISCO-IETF-PW-MPLS-MIB

For more information about MIB support on a Cisco 7600 series router, refer to the *Cisco 7600 Series Internet Router MIB Specifications Guide*, at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/products_mib_quick_reference_book09186a00807f69b0.html

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

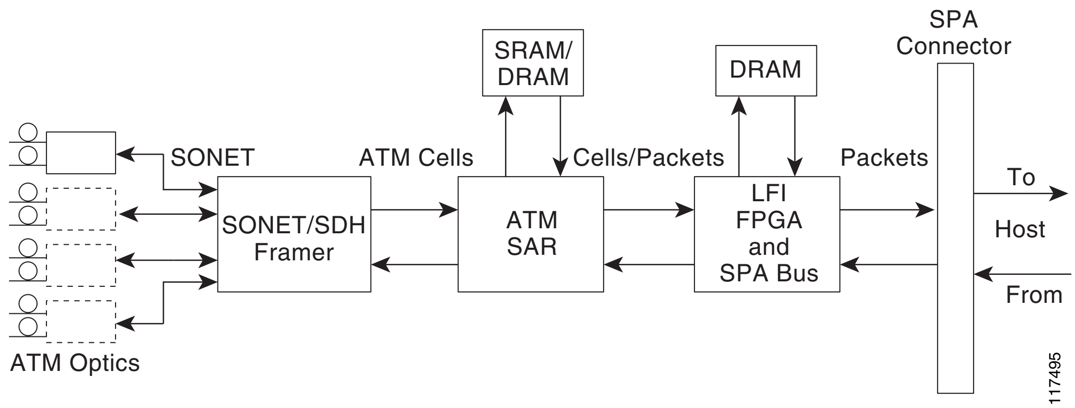
To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

SPA Architecture

This section provides an overview of the data path for the ATM SPAs, for use in troubleshooting and monitoring. Figure 6-6 shows the data path for ATM traffic as it travels between the ATM optical connectors on the front panel of the ATM SPA to the backplane connector that connects the SPA to the SIP.

Figure 6-6 ATM SPA Data Architecture



Path of Cells in the Ingress Direction

The following steps describe the path of an ingress cell as it is received from the ATM network and converted to a data packet before transmission through the SIP to the router's processors for switching, routing, or further processing:

1. The SONET/SDH framer device receives incoming cells on a per-port basis from the SPA's optical circuitry. (The ATM SPA supports 1, 2, or 4 optical ports, depending on the model of SPA.)
2. The SONET/SDH framer removes the SONET overhead information, performs any necessary clock and data recovery, and processes any SONET/SDH alarms that might be present. The framer then extracts the 53-byte ATM cells from the data stream and forwards each cell to the ATM segmentation and reassembly (SAR) engine.
3. The SAR engine receives the cells from the framer and reassembles them into the original packets, temporarily storing them in a per-port receive buffer until they can be forwarded to the LFI field-programmable gate array (FPGA). The SAR engine discards any packets that have been corrupted in transit.
4. The LFI FPGA receives the packets from the SAR engine and forwards them to the host processor for further routing, switching, or additional processing. The FPGA also performs LFI reassembly as needed, and collects the traffic statistics for the packets that it passes.

Path of Packets in the Egress Direction

The following steps describe the path of an egress packet as the SPA receives it from the router through the SIP and converts it to ATM cells for transmission on the ATM network:

1. The LFI FPGA receives the packets from the host processor and stores them in its packet buffers until the SAR engine is ready to receive them. The FPGA also performs any necessary LFI processing on the packets before forwarding them to the SAR engine. The FPGA also collects the traffic statistics for the packets that it passes.
2. The SAR engine receives the packets from the FPGA and supports multiple CBWFQ queues to store the packets until they can be fully segmented. The SAR engine performs the necessary WRED queue admission and CBWFQ QoS traffic scheduling on its queues before segmenting the packets into ATM cells and shaping the cells into the SONET/SDH framer.
3. The SONET/SDH framer receives the packets from the SAR engine and inserts each cell into the SONET data stream, adding the necessary clocking, SONET overhead, and alarm information. The framer then outputs the data stream out the appropriate optical port.
4. The optical port conveys the optical data onto the physical layer of the ATM network.

Displaying the SPA Hardware Type

To verify the SPA hardware type that is installed in your Cisco 7600 series router, use the **show interfaces**, **show diag**, or **show controllers** commands. A number of other **show** commands also provide information about the SPA hardware.

Table 6-2 shows the hardware description that appears in the **show interfaces** and **show diag** command output for each type of ATM SPA that is supported on the Cisco 7600 series router:

Table 6-2 ATM SPA Hardware Descriptions in show Commands

SPA	Description in show interfaces Command	Description in show diag Command
SPA-2XOC3-ATM	"Hardware is SPA-2XOC3-ATM"	"SPA-2XOC3-ATM (0x046E)"
SPA-4XOC3-ATM	"Hardware is SPA-4XOC3-ATM"	"SPA-4XOC3-ATM (0x3E1)"
SPA-1XOC12-ATM	"Hardware is SPA-1XOC12-ATM"	"SPA-1XOC12-ATM (0x03E5)"
SPA-1XOC48-ATM	"Hardware is SPA-1XOC48-ATM"	"SPA-1XOC48-ATM (0x3E6)"

Example of the show interfaces Command

The following example shows output from the **show interfaces atm** command on a Cisco 7600 series router with an ATM SPA installed in the first subslot of a SIP that is installed in slot 5:

```
Router# show interfaces atm 5/0/0

ATM5/0/0 is up, line protocol is up
  Hardware is SPA-4XOC3-ATM, address is 000d.2959.d780 (bia 000d.2959.d78a)
  MTU 4470 bytes, sub MTU 4470, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ATM, loopback not set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 1 current VCCs
  VC idle disconnect time: 300 seconds
  0 carrier transitions
  Last input 00:00:09, output 00:00:09, output hang never
  Last clearing of "show interface" counters 00:01:26
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     5 packets input, 540 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     5 packets output, 720 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Note The value for "packets output" in the default version of the **show interfaces atm** command includes the bytes used for ATM AAL5 padding, trailer and ATM cell header. To see the packet count without the padding, header, and trailer information, use the **show interfaces atm statistics** or **show atm pvc** commands.

Example of the show diag Command

The following example shows output from the **show diag** command on a Cisco 7600 series router with two ATM SPAs installed in a Cisco 7600 SIP-400 that is installed in slot 4:

```
Router# show diag 4

Slot 4: Logical_index 8
        4-adapter SIP-400 controller
        Board is analyzed ipc ready
        HW rev 0.300, board revision 08
        Serial Number:  Part number: 73-8272-03

        Slot database information:
        Flags: 0x2004   Insertion time: 0x1961C (01:16:54 ago)

        Controller Memory Size:
                384 MBytes CPU Memory
                128 MBytes Packet Memory
                512 MBytes Total on Board SDRAM
        IOS (tm) cwlc Software (sip1-DW-M), Released Version 12.2(17)SX [BLD-sipedon2 107]

        SPA Information:
        subslot 4/0: SPA-4XOC3-ATM (0x3E1), status: ok
        subslot 4/1: SPA-1XOC12-ATM (0x3E5), status: ok
```

Example of the show controllers Command

The following example shows output from the **show controllers atm** command on a Cisco 7600 series router with an ATM SPA installed in the second subslot of a SIP that is installed in slot 5:

```
Router# show controllers atm 5/1/0

Interface ATM5/1/0 (SPA-4XOC3-ATM[4/0]) is up
 Framing mode: SONET OC3 STS-3c

SONET Subblock:
SECTION
  LOF = 0          LOS    = 0                            BIP(B1) = 603
LINE
  AIS = 0          RDI    = 2          FEBE = 2332       BIP(B2) = 1018
PATH
  AIS = 0          RDI    = 1          FEBE = 28         BIP(B3) = 228
  LOP = 0          NEWPTR = 0          PSE  = 1          NSE     = 2

Active Defects: None
Active Alarms:  None
Alarm reporting enabled for: SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA

ATM framing errors:
  HCS (correctable):   0
  HCS (uncorrectable): 0

APS
 not configured

PATH TRACE BUFFER : STABLE

BER thresholds:  SF = 10e-3  SD = 10e-6
TCA thresholds:  B1 = 10e-6  B2 = 10e-6  B3 = 10e-6

  Clock source:  line
```

CHAPTER 7

Configuring the ATM SPAs

This chapter provides information about configuring the ATM SPAs on the Cisco 7600 series router. It includes the following sections:

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications that correspond to your Cisco IOS software release.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Configuration Tasks

This section describes the most common configurations for the ATM SPAs on a Cisco 7600 series router. It contains procedures for the following configurations:

Required Configuration Tasks

The ATM SPA interface must be initially configured with an IP address to allow further configuration. Some of the required configuration commands implement default values that might or might not be appropriate for your network. If the default value is correct for your network, then you do not need to configure the command. To perform the basic configuration of each interface, use the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-if)# **ip address** *address mask* [**secondary**]	(Optional in some configurations) Assigns the specified IP address and subnet mask to the interface. Repeat the command with the optional **secondary** keyword to assign additional, secondary IP addresses to the port.
Step 3	Router(config-if)# **description** *string*	(Optional) Assigns an arbitrary string, up to 80 characters long, to the interface. This string can identify the purpose or owner of the interface, or any other information that might be useful for monitoring and troubleshooting.
Step 4	Router(config-if)# **no shutdown**	Enables the interface.
	Note Repeat Step 1 through Step 4 for each port on the ATM SPA to be configured.	
Step 5	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Specifying the Interface Address on a SPA

Two ATM SPAs can be installed in a SIP. SPA interface ports begin numbering with "0" from left to right. Single-port SPAs use only the port number 0. To configure or monitor SPA interfaces, you need to specify the physical location of the SIP, SPA, and interface in the CLI. The interface address format is *slot/subslot/port*, where:

- *slot*—Specifies the chassis slot number in the Cisco 7600 series router where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA.

The following example shows how to specify the first interface (0) on a SPA installed in the first subslot of a SIP (0) installed in chassis slot 3:

```
Router(config)# interface serial 3/0/0
```

This command shows a serial SPA as a representative example, however the same *slot/subslot/port* format is similarly used for other SPAs (such as ATM and POS) and other non-channelized SPAs.

For more information about identifying slots and subslots, see the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section on page 4-2.

Modifying the Interface MTU Size

The maximum transmission unit (MTU) values might need to be reconfigured from their defaults on the ATM SPAs to match the values used in your network.

Interface MTU Configuration Guidelines

When configuring the interface MTU size on an ATM SPA, consider the following guidelines.

The Cisco IOS software supports several types of configurable MTU options at different levels of the protocol stack. You should ensure that all MTU values are consistent to avoid unnecessary fragmentation of packets. These MTU values are the following:

- Interface MTU—Configured on a per-interface basis and defines the maximum packet size (in bytes) that is allowed for traffic received on the network. The ATM SPA checks traffic coming in from the network and drops packets that are larger than this maximum value. Because different types of Layer 2 interfaces support different MTU values, choose a value that supports the maximum possible packet size that is possible in your particular network topology.
- IP MTU—Configured on a per-interface or per-subinterface basis and determines the largest maximum IP packet size (in bytes) that is allowed on the IP network without being fragmented. If an IP packet is larger than the IP MTU value, the ATM SPA fragments it into smaller IP packets before forwarding it on to the next hop.
- Multiprotocol Label Switching (MPLS) MTU—Configured on a per-interface or per-subinterface basis and defines the MTU value for packets that are tagged with MPLS labels or tag headers. When an IP packet that contains MPLS labels is larger than the MPLS MTU value, the ATM SPA fragments it into smaller IP packets. When a non-IP packet that contains MPLS labels is larger than the MPLS MTU value, the ATM SPA drops it.

All devices on a particular physical medium must have the same MPLS MTU value to allow proper MPLS operation. Because MPLS labels are added on to the existing packet and increase the packet's size, choose appropriate MTU values so as to avoid unnecessarily fragmenting MPLS-labeled packets.

If the IP MTU or MPLS MTU values are currently the same size as the interface MTU, changing the interface MTU size also automatically sets the IP MTU or MPLS MTU values to the new value. Changing the interface MTU value does not affect the IP MTU or MPLS MTU values if they are not currently set to the same size as the interface MTU.

Different encapsulation methods and the number of MPLS MTU labels add additional overhead to a packet. For example, Subnetwork Access Protocol (SNAP) encapsulation adds an 8-byte header, IEEE 802.1Q encapsulation adds a 2-byte header, and each MPLS label adds a 4-byte header. Consider the maximum possible encapsulations and labels that are to be used in your network when choosing the MTU values.

Tip The MTU values on the local ATM SPA interfaces must match the values being used in the ATM network and remote ATM interface. Changing the MTU values on an ATM SPA does not reset the local interface, but be aware that other platforms and ATM SPAs do reset the link when the MTU value changes. This could cause a momentary interruption in service, so we recommend changing the MTU value only when the interface is not being used.

Note The interface MTU value on the ATM SPA also determines which packets are recorded as "giants" in the **show interfaces atm** command. The interface considers a packet to be a giant packet when it is more than 24 bytes larger than the interface MTU size. For example, if using an MTU size of 1500 bytes, the interface increments the giants counter when it receives a packet larger than 1524 bytes.

Interface MTU Configuration Task

To change the MTU values on the ATM SPA interfaces, use the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-if)# **mtu** *bytes*	(Optional) Configures the maximum transmission unit (MTU) size for the interface. The valid range for *bytes* is from 64 to 9216 bytes, with a default of 4470 bytes. As a general rule, do not change the MTU value unless you have a specific application need to do so. **Note** If the IP MTU or MPLS MTU values are currently the same size as the interface MTU, changing the interface MTU size also automatically sets the IP MTU or MPLS MTU values to the same value.
Step 3	Router(config-if)# **ip mtu** *bytes*	(Optional) Configures the MTU value, in bytes, for IP packets on this interface. The valid range for an ATM SPA is 64 to 9288, with a default value equal to the MTU value configured in Step 2.

	Command or Action	Purpose
Step 4	Router(config-if)# **mpls mtu** *bytes*	(Optional) Configures the MTU value, in bytes, for MPLS-labeled packets on this interface. The valid range for an ATM SPA is 64 to 9216 bytes, with a default value equal to the MTU value configured in Step 2.
	Note Repeat Step 1 through Step 4 for each interface port on the ATM SPA to be configured.	
Step 5	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying the MTU Size

To verify the MTU sizes for an interface, use the **show interface**, **show ip interface**, and **show mpls interface** commands, as in the following examples:

```
Router# show interface atm 4/1/0

ATM4/1/0 is up, line protocol is up
  Hardware is SPA-4XOC3-ATM, address is 000d.2959.d5ca (bia 000d.2959.d5ca)
  MTU 4470 bytes, sub MTU 4470, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ATM, loopback not set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 0 current VCCs
  VC idle disconnect time: 300 seconds
  0 carrier transitions
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out

Router# show ip interface atm 4/1/0

ATM4/1/0 is up, line protocol is up
  Internet address is 200.1.0.2/24
  Broadcast address is 255.255.255.255
  Address determined by non-volatile memory
  MTU is 4470 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Multicast reserved groups joined: 224.0.0.9
  Outgoing access list is not set
  Inbound  access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is enabled
  IP fast switching on the same interface is disabled
```

```
  IP Flow switching is disabled
  IP Feature Fast switching turbo vector
  IP Null turbo vector
  VPN Routing/Forwarding "vpn2600-2"
  IP multicast fast switching is enabled
  IP multicast distributed fast switching is disabled
  IP route-cache flags are Fast, CEF
  Router Discovery is disabled
  IP output packet accounting is disabled
  IP access violation accounting is disabled
  TCP/IP header compression is disabled
  RTP/IP header compression is disabled
  Probe proxy name replies are disabled
  Policy routing is disabled
  Network address translation is disabled
  WCCP Redirect outbound is disabled
  WCCP Redirect exclude is disabled
  BGP Policy Mapping is disabled

Router# show mpls interface atm 4/1/0 detail

Interface ATM3/0:
        IP labeling enabled (ldp)
        LSP Tunnel labeling not enabled
        MPLS operational
        MPLS turbo vector
        MTU = 4470
        ATM labels: Label VPI = 1
                Label VCI range = 33 - 65535
                Control VC = 0/32
```

To view the maximum possible size for datagrams passing out the interface using the configured MTU value, use the **show atm interface atm** command:

```
Router# show atm interface atm 4/1/0
   Interface ATM4/1/0:
   AAL enabled: AAL5, Maximum VCs: 4096, Current VCCs: 2
   Maximum Transmit Channels: 0
   Max. Datagram Size: 4528
   PLIM Type: SONET - 155000Kbps, TX clocking: LINE
   Cell-payload scrambling: ON
   sts-stream scrambling: ON
   8359 input, 8495 output, 0 IN fast, 0 OUT fast, 0 out drop
   Avail bw = 155000
   Config. is ACTIVE
```

Creating a Permanent Virtual Circuit

To use a permanent virtual circuit (PVC), configure the PVC in both the router and the ATM switch. PVCs remain active until the circuit is removed from either configuration. To create a PVC on the ATM interface and enter interface ATM VC configuration mode, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot*/*subslot*/*port* or Router(config)# **interface atm** *slot*/*subslot*/*port.subinterface*	Enters interface or subinterface configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-if)# **ip address** *address mask*	Assigns the specified IP address and subnet mask to the interface or subinterface.
Step 3	Router(config-if)# **atm tx-latency** *milliseconds*	(Optional) Configures the default transmit latency for VCs on this ATM SPA interface. The valid range for *milliseconds* is from 1 to 200, with a default of 100 milliseconds.
Step 4	Router(config-if)# **pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi*/*vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the VC to exclusively carry ILMI protocol traffic (default). • **qsaal**—(Optional) Configures the VC to exclusively carry QSAAL protocol traffic.

Note When using the **pvc** command, remember that the *vpi/vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi/vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.

	Command or Action	Purpose
Step 5	Router(config-if-atm-vc)# **protocol** *protocol* {*protocol-address* | **inarp**} [[**no**] **broadcast**]	Configures the PVC for a particular protocol and maps it to a specific *protocol-address*. • *protocol*—Typically set to either **ip** or **ppp**, but other values are possible. • *protocol-address*—Destination address or virtual interface template for this PVC (if appropriate for the *protocol*). • **inarp**—Specifies that the PVC uses Inverse ARP to determine its address. • [**no**] **broadcast**—(Optional) Specifies that this mapping should (or should not) be used for broadcast packets.
Step 6	Router(config-if-atm-vc)# **inarp** *minutes*	(Optional) If using Inverse ARP, configures how often the PVC transmits Inverse ARP requests to confirm its address mapping. The valid range is 1 to 60 minutes, with a default of 15 minutes.
Step 7	Router(config-if-atm-vc)# **encapsulation aal5snap**	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type. The default and only supported type is **aal5snap**.
Step 8	Router(config-if-atm-vc)# **tx-limit** *buffers*	(Optional) Specifies the number of transmit buffers for this VC. The valid range is from 1 to 57343, with a default value that is based on the current VC line rate and on the latency value that is configured with the **atm tx-latency** command.
	Note Repeat Step 4 through Step 8 for each PVC to be configured on this interface.	
Step 9	Router(config-if-atm-vc)# **end**	Exits ATM VC configuration mode and returns to privileged EXEC mode.

Verifying a PVC Configuration

To verify the configuration of a particular PVC, use the **show atm pvc** command:

```
Router# show atm pvc 1/100

ATM3/0/0: VCD: 1, VPI: 1, VCI: 100
UBR, PeakRate: 149760
AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC status: Not Managed
ILMI VC status: Not Managed
InARP frequency: 15 minutes(s)
Transmit priority 6
InPkts: 94964567, OutPkts: 95069747, InBytes: 833119350, OutBytes: 838799016
InPRoc: 1, OutPRoc: 1, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 94964566, OutAS: 95069746
InPktDrops: 0,  OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0, LengthViolation: 0, CPIErrors: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
```

```
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
OAM cells sent: 0
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutRDI: 0
F4 OutEndloop: 0, F4 OutSegloop: 0, F4 OutRDI: 0
OAM cell drops: 0
Status: UP

VC 1/100 doesn't exist on 7 of 8 ATM interface(s)
```

Tip To verify the configuration and current status of all PVCs on a particular interface, you can also use the **show atm vc interface atm** command.

Creating a PVC on a Point-to-Point Subinterface

Use point-to-point subinterfaces to provide each pair of routers with its own subnet. When you create a PVC on a point-to-point subinterface, the router assumes it is the only point-to-point PVC that is configured on the subinterface, and it forwards all IP packets with a destination IP address in the same subnet to this VC. To configure a point-to-point PVC, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port.subinterface* **point-to-point**	Creates the specified point-to-point subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode.
Step 2	Router(config-subif)# **ip address** *address mask*	Assigns the specified IP address and subnet mask to this subinterface.
Step 3	Router(config-subif)# **pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi*/*vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the PVC to use ILMI encapsulation (default). • **qsaal**—(Optional) Configures the PVC to use QSAAL encapsulation.

Note When using the **pvc** command, remember that the *vpi*/*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi*/*vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.

	Command or Action	Purpose
Step 4	Router(config-if-atm-vc)# **protocol** *protocol protocol-address* [[**no**] **broadcast**]	Configures the PVC for a particular protocol and maps it to a specific *protocol-address*. • *protocol*—Typically set to **ppp** for point-to-point subinterfaces, but other values are possible. • *protocol-address*—Destination address or virtual template interface for this PVC (as appropriate for the specified *protocol*). • [**no**] **broadcast**—(Optional) Specifies that this mapping should (or should not) be used for broadcast packets. The **protocol** command also has an **inarp** option, but this option is not meaningful on point-to-point PVCs that use a manually configured address.
Step 5	Router(config-if-atm-vc)# **encapsulation aal5snap**	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type. The default and only supported type is **aal5snap**.
	Note Repeat Step 1 through Step 5 for each point-to-point subinterface to be configured on this ATM SPA.	
Step 6	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying a Point-to-Point PVC Configuration

To verify the configuration of a particular PVC, use the **show atm pvc** command:

```
Router# show atm pvc 3/12

ATM3/1/0.12: VCD: 3, VPI: 3, VCI: 12
UBR, PeakRate: 149760
AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC status: Not Managed
ILMI VC status: Not Managed
InARP frequency: 15 minutes(s)
Transmit priority 6
InPkts: 3949645, OutPkts: 3950697, InBytes: 28331193, OutBytes: 28387990
InPRoc: 1, OutPRoc: 1, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 3949645, OutAS: 3950697
InPktDrops: 0,  OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0, LengthViolation: 0, CPIErrors: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
OAM cells sent: 0
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutRDI: 0
F4 OutEndloop: 0, F4 OutSegloop: 0, F4 OutRDI: 0
OAM cell drops: 0
Status: UP
```

Tip To verify the configuration and current status of all PVCs on a particular interface, you can also use the **show atm vc interface atm** command.

Configuring a PVC on a Multipoint Subinterface

Creating a multipoint subinterface allows you to create a point-to-multipoint PVC that can be used as a broadcast PVC for all multicast requests. To create a PVC on a multipoint subinterface, use the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot*/*subslot*/*port.subinterface* **multipoint**	Creates the specified point-to-multipoint subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode.
Step 2	Router(config-subif)# **ip address** *address mask*	Assigns the specified IP address and subnet mask to this subinterface.
Step 3	Router(config-subif)# **no ip directed-broadcast**	(Optional) Disables the forwarding of IP directed broadcasts, which are sometimes used in denial of service (DOS) attacks.
Step 4	Router(config-subif)# **pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi*/*vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the PVC to use ILMI encapsulation (default). • **qsaal**—(Optional) Configures the PVC to use QSAAL encapsulation.
	Note When using the **pvc** command, remember that the *vpi*/*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi*/*vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.	
Step 5	Router(config-if-atm-vc)# **protocol** *protocol* {*protocol-address* | **inarp**} **broadcast**	Configures the PVC for a particular protocol and maps it to a specific *protocol-address*. • *protocol*—Typically set to **ip** for multipoint subinterfaces, but other values are possible. • *protocol-address*—Destination address or virtual template interface for this PVC (if appropriate for the *protocol*). • **inarp**—Specifies that the PVC uses Inverse ARP to determine its address. • **broadcast**— Specifies that this mapping should be used for multicast packets.

	Command or Action	Purpose
Step 6	Router(config-if-atm-vc)# **inarp** *minutes*	(Optional) If using Inverse ARP, configures how often the PVC transmits Inverse ARP requests to confirm its address mapping. The valid range is 1 to 60 minutes, with a default of 15 minutes.
Step 7	Router(config-if-atm-vc)# **encapsulation aal5snap**	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type. The default and only supported type is **aal5snap**.
	Note Repeat Step 1 through Step 7 for each multipoint subinterface to be configured on this ATM SPA.	
Step 8	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying a Multipoint PVC Configuration

To verify the configuration of a particular PVC, use the **show atm pvc** command:

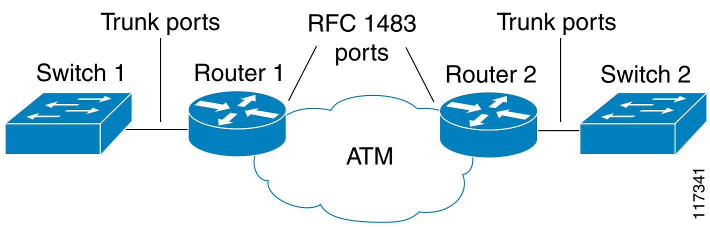
```
Router# show atm pvc 1/120

ATM3/1/0.120: VCD: 1, VPI: 1, VCI: 120
UBR, PeakRate: 149760
AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC status: Not Managed
ILMI VC status: Not Managed
InARP frequency: 15 minutes(s)
Transmit priority 6
InPkts: 1394964, OutPkts: 1395069, InBytes: 1833119, OutBytes: 1838799
InPRoc: 1, OutPRoc: 1, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 94964, OutAS: 95062
InPktDrops: 0,  OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0, LengthViolation: 0, CPIErrors: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
OAM cells sent: 0
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutRDI: 0
F4 OutEndloop: 0, F4 OutSegloop: 0, F4 OutRDI: 0
OAM cell drops: 0
Status: UP
```

Note To verify the configuration and current status of all PVCs on a particular interface, you can also use the **show atm vc interface atm** command.

Configuring RFC 1483 Bridging for PVCs

RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer* 5, specifies the implementation of point-to-point bridging of Layer 2 protocol data units (PDUs) from an ATM interface. Figure 7-1 shows an example in which the two routers receive VLANs over their respective trunk links and then forward that traffic out through the ATM interfaces into the ATM cloud.

Figure 7-1 Example of RFC 1483 Bridging Topology



Note RFC 1483 has been updated and superseded by RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*.

RFC 1483 Bridging for PVCs Configuration Guidelines

When configuring RFC 1483 bridging for PVCs, consider the following guidelines:

- PVCs must use AAL5 Subnetwork Access Protocol (SNAP) encapsulation.
- To use the Virtual Trunking Protocol (VTP), ensure that each main interface has a subinterface that has been configured for the management VLANs (VLAN 1 and VLANs 1002 to 1005). VTP is not supported on bridged VCs on a Cisco 7600 SIP-200.
- RFC 1483 bridging in a switched virtual circuit (SVC) environment is not supported.
- The 1-Port OC-48c/STM-16 ATM SPA does not support RFC 1483 bridging.

RFC 1483 Bridging for PVCs Configuration Task

To configure RFC 1483 bridging for PVCs, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port.subinterface* **point-to-point**	(Optional) Creates the specified point-to-point subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode. **Note** Although it is most common to create the PVCs on subinterfaces, you can also omit this step to create the PVCs for RFC 1483 bridging on the main interface.
Step 2	Router(config-subif)# **pvc** [*name*] *vpi/vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi/vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the PVC to use ILMI encapsulation (default). • **qsaal**—(Optional) Configures the PVC to use QSAAL encapsulation.

	Command or Action	Purpose
Step 3	Router(config-if-atm-vc)# **bridge-domain** *vlan-id* [**access** | **dot1q** *tag* | **dot1q-tunnel**] [**ignore-bpdu-pid**] | {**pvst-tlv** *CE-vlan*} [**increment**] [**split-horizon**]	Binds the PVC to the specified *vlan-id*. You can optionally specify the following keywords: • **dot1q**—(Optional) Includes the IEEE 802.1Q tag, which preserves the VLAN ID and class of service (CoS) information across the ATM cloud. • **dot1q-tunnel**—(Optional) Enables tunneling of IEEE 802.1Q VLANs over the same link. See the "Configuring RFC 1483 Bridging for PVCs with IEEE 802.1Q Tunneling" section on page 7-16. • **ignore-bpdu-pid**—(Optional) Ignores bridge protocol data unit (BPDU) packets, to allow interoperation with ATM customer premises equipment (CPE) devices that do not distinguish BPDU packets from data packets. Without this keyword, IEEE BPDUs are sent out using a PID of 0x00-0E, which complies with RFC 1483. With this keyword, IEEE BPDUs are sent out using a PID of 0x00-07, which is normally reserved for RFC 1483 data. • **pvst-tlv**—When transmitting, the **pvst-tlv** keyword translates PVST+ BPDUs into IEEE BPDUs. When receiving, the **pvst-tlv** keyword translates IEEE BPDUs into PVST+ BPDUs. • **split-horizon**—(Optional) Enables RFC 1483 split horizon mode to globally prevent bridging between PVCs in the same VLAN.
Step 4	Router(config-if-atm-vc)# **encapsulation aal5snap**	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type. The default and only supported type is **aal5snap**.
	Note Repeat Step 1 through Step 4 for each interface on the ATM SPA to be configured.	
Step 5	Router(config-if-atm-vc)# **end**	Exits ATM VC configuration mode and returns to privileged EXEC mode.

Verifying the RFC 1483 Bridging Configuration

To verify the RFC 1483 bridging configuration and status, use the **show interface atm** command:

```
Router# show interface atm 6/1/0.3

ATM6/1/0.3 is up, line protocol is up
  Hardware is SPA-4XOC3-ATM
  Internet address is 10.10.10.13/24
  MTU 4470 bytes, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ATM
  5 packets input, 566 bytes
  5 packets output, 566 bytes
  1445 OAM cells input, 1446 OAM cells output
```

Configuring Layer 2 Protocol Tunneling Topology

To enable BPDU translation for the Layer 2 Protocol Tunneling (L2PT) topologies, use the following command line:

bridge-domain *PE vlan* **dot1q-tunnel ignore-bpdu-pid pvst-tlv** *CE vlan*

Configuring RFC 1483 Bridging for PVCs with IEEE 802.1Q Tunneling

RFC 1483 bridging (see the "Configuring RFC 1483 Bridging for PVCs" section on page 7-13) can also include IEEE 802.1Q tunneling, which allows service providers to aggregate multiple VLANs over a single VLAN, while still keeping the individual VLANs segregated and preserving the VLAN IDs for each customer. This tunneling simplifies traffic management for the service provider, while keeping the customer networks secure.

Also, the IEEE 802.1Q tunneling is configured only on the service provider routers, so it does not require any additional configuration on the customer-side routers. The customer side is not aware of the configuration.

Note For complete information on IEEE 802.1Q tunneling on a Cisco 7600 series router, see the *Cisco 7600 Series Cisco IOS Software Configuration Guide*, *12.2SX* at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/products_configuration_guide_chapter09186a0080160eba.html

Note RFC 1483 has been updated and superseded by RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*.

RFC 1483 Bridging for PVCs with IEEE 802.1Q Tunneling Configuration Guidelines

When configuring RFC 1483 bridging for PVCs with IEEE 802.1Q tunneling, consider the following guidelines:

- Customer equipment must be configured for RFC 1483 bridging with IEEE 802.1Q tunneling using the **bridge-domain dot1q** ATM VC configuration command. See the "Configuring RFC 1483 Bridging for PVCs" section on page 7-13 for more information.
- PVCs must use AAL5 encapsulation.
- RFC 1483 bridged PVCs must terminate on the ATM SPA, and the traffic forwarded over this bridged connection to the edge must be forwarded through an Ethernet port.
- To use the Virtual Trunking Protocol (VTP), each main interface should have a subinterface that has been configured for the management VLANs (VLANs 1 and 1002–1005).
- RFC 1483 bridging in a switched virtual circuit (SVC) environment is not supported.

RFC 1483 Bridging for PVCs with IEEE 802.1Q Tunneling Configuration Task

To configure RFC 1483 bridging for PVCs with IEEE 802.1Q tunneling, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port.subinterface* **point-to-point**	(Optional) Creates the specified point-to-point subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode. **Note** Although it is most common to create the PVCs on subinterfaces, you can also omit this step to create the PVCs for RFC 1483 bridging on the main interface.
Step 2	Router(config-subif)# **pvc** [*name*] *vpi/vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi/vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the PVC to use ILMI encapsulation (default). • **qsaal**—(Optional) Configures the PVC to use QSAAL encapsulation.
	Note When using the **pvc** command, remember that the *vpi/vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi/vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.	
Step 3	Router(config-if-atm-vc)# **bridge-domain** *vlan-id* **dot1q-tunnel**	Binds the PVC to the specified *vlan-id* and enables the use of IEEE 802.1Q tunneling on the PVC. This preserves the VLAN ID information across the ATM cloud.
Step 4	Router(config-if-atm-vc)# **encapsulation aal5snap**	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type. The default and only supported type is **aal5snap**.
	Note Repeat Step 1 through Step 4 for each interface on the ATM SPA to be configured.	
Step 5	Router(config-if-atm-vc)# **end**	Exits ATM VC configuration mode and returns to privileged EXEC mode.

Verifying the RFC 1483 for PVCs Bridging with IEEE 802.1Q Tunneling Configuration

To verify the IEEE 802.1Q tunneling on an ATM SPA, use the **show 12-protocol-tunnel** command:

```
Router# show l2protocol-tunnel

COS for Encapsulated Packets: 5

Port     Protocol  Shutdown  Drop      Encapsulation Decapsulation Drop
                   Threshold Threshold Counter       Counter       Counter
-------- --------  --------- --------- ------------- ------------- -------------
Gi4/2     cdp           ----      ----             0             0             0
          stp           ----      ----             0             0             0
          vtp           ----      ----             0             0             0
ATM6/2/1 cdp           ----      ----           n/a           n/a           n/a
          stp           ----      ----           n/a           n/a           n/a
          vtp           ----      ----           n/a           n/a           n/a
```

Note The counters in the output of the **show l2protocol-tunnel** command are not applicable for ATM interfaces when IEEE 802.1Q tunneling is enabled.

Use the following command to display the interfaces that are configured with an IEEE 802.1Q tunnel:

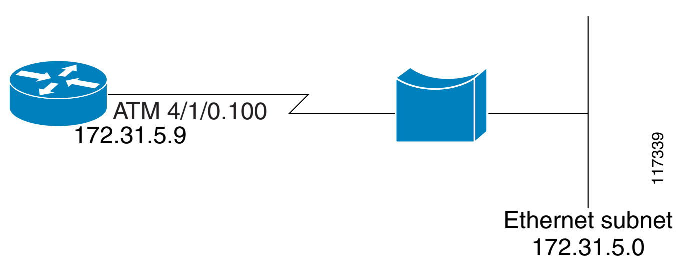
```
Router# show dot1q-tunnel
LAN Port(s)
-----------
Gi4/2

ATM Port(s)
-----------
ATM6/2/1
```

Configuring ATM RFC 1483 Half-Bridging

The ATM SPA supports ATM RFC 1483 half-bridging, which routes IP traffic from a stub-bridged Ethernet LAN over a bridged RFC 1483 ATM interface, without using integrated routing and bridging (IRB). This allows bridged traffic that terminates on an ATM PVC to be routed on the basis of the destination IP address.

For example, Figure 7-2 shows a remote bridged Ethernet network connecting to a routed network over a device that bridges the Ethernet LAN to the ATM interface.

Figure 7-2 ATM RFC 1483 Half-Bridging



When half-bridging is configured, the ATM interface receives the bridged IP packets and routes them according to each packet's IP destination address. Similarly, when packets are routed to this ATM PVC, it then forwards them out as bridged packets on its bridge connection.

This use of a stub network topology offers better performance and flexibility over integrated routing and bridging (IRB). This also helps to avoid a number of issues such as broadcast storms and security risks.

In particular, half-bridging reduces the potential security risks that are associated with normal bridging configurations. Because the ATM interface allocates a single virtual circuit (VC) to a subnet (which could be as small as a single IP address), half-bridging limits the size of the nonsecured network that can be allowed access to the larger routed network. This makes half-bridging configurations ideally suited for customer access points, such digital subscriber lines (DSL).

Note RFC 1483 has been updated and superseded by RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*. However, to avoid confusion, this document continues to use the previously-used terminology of "RFC 1483 ATM half-bridging."

To configure a point-to-multipoint ATM PVC for ATM half-bridging, use the configuration task in the following section.

Note Use the following configuration task when you want to configure point-to-multipoint PVCs for half-bridging operation. Use the configuration task in the "Configuring ATM Routed Bridge Encapsulation" section on page 7-21 to configure a point-to-point PVC for similar functionality.

ATM RFC 1483 Half-Bridging Configuration Guidelines

When configuring ATM RFC 1483 half-bridging, consider the following guidelines:

- Supports only IP traffic and access lists.
- Supports only fast switching and process switching.
- Supports only PVCs that are configured on multipoint subinterfaces. SVCs are not supported for half-bridging.
- A maximum of one PVC can be configured for half-bridging on each subinterface. Other PVCs can be configured on the same subinterface, as long as they are not configured for half-bridging as well.
- The same PVC cannot be configured for both half-bridging and full bridging.

ATM RFC 1483 Half-Bridging Configuration Task

To configure ATM RFC 1483 half-bridging, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port.subinterface* **multipoint**	Creates the specified point-to-point subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode.
Step 2	Router(config-subif)# **ip address** *address mask* [**secondary**]	Assigns the specified IP address and subnet mask to this subinterface. This IP address should be on the same subnet as the remote bridged network (the Ethernet network).

	Command or Action	Purpose
Step 3	Router(config-subif)# **pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi*/*vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the PVC to use ILMI encapsulation (default). • **qsaal**—(Optional) Configures the PVC to use QSAAL encapsulation.
	Note When using the **pvc** command, remember that the *vpi*/*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi*/*vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.	
Step 4	Router(config-if-atm-vc)# **encapsulation aal5snap bridge**	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type, and specifies that half-bridging should be used.
Step 5	Router(config-if-atm-vc)# **end**	Exits ATM VC configuration mode and returns to privileged EXEC mode.

Verifying the ATM RFC 1483 Half-Bridging Configuration

To verify the ATM RFC 1483 half-bridging configuration, use the **show atm vc** command:

```
Router# show atm vc 20

ATM4/0/0.20: VCD: 20, VPI: 1, VCI: 20
UBR, PeakRate: 149760
AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 0 second(s)
InARP frequency: 15 minutes(s), 1483-half-bridged-encap
Transmit priority 6
InPkts: 2411, OutPkts: 2347, InBytes: 2242808, OutBytes: 1215746
InPRoc: 226, OutPRoc: 0
InFast: 0, OutFast: 0, InAS: 2185, OutAS: 2347
InPktDrops: 1,  OutPktDrops: 0
InByteDrops: 0, OutByteDrops: 0
CrcErrors: 139, SarTimeOuts: 0, OverSizedSDUs: 0, LengthViolation: 0, CPIErrors: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
OAM cells sent: 0
Status: UP
```

Configuring ATM Routed Bridge Encapsulation

The ATM SPAs support ATM Routed Bridge Encapsulation (RBE), which is similar in functionality to RFC 1483 ATM half-bridging, except that ATM half-bridging is configured on a point-to-multipoint PVC, while RBE is configured on a point-to-point PVC (see the "Configuring ATM RFC 1483 Half-Bridging" section on page 7-18).

Note The 1-Port OC-48c/STM-16 ATM SPA does not support RBE.

Use the following configuration task to configure a point-to-point subinterface and PVC for RBE bridging.

Note RFC 1483 has been updated and superseded by RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*.

ATM Routed Bridge Encapsulation Configuration Guidelines

When configuring ATM RBE, consider the following guidelines:

- Supported only on ATM SPAs in a Cisco 7600 SIP-200. RBE is not supported when using a Cisco 7600 SIP-400.
- Supports only AAL5SNAP encapsulation.
- Supports only IP access lists, not MAC-layer access lists.
- Supports only fast switching and process switching.
- Supports distributed Cisco Express Forwarding (dCEF).
- Supports only PVCs on point-to-point subinterfaces. SVCs are not supported for half-bridging.
- The **bridge-domain** command cannot be used on any PVC that is configured for RBE, because an RBE PVC acts as the termination point for bridged packets.
- The **atm bridge-enable** command, which was used in previous releases on other ATM interfaces, is not supported on ATM SPA interfaces.
- The IS-IS protocol is not supported with point-to-point PVCs that are configured for RBE bridging.

RBE Configuration Limitation Supports Only One Remote MAC Address

On the Cisco 7600 series router with a Supervisor Engine 720 or Route Switch Processor 720 (RSP720) and the following SPA, an ATM PVC with an RBE configuration can send packets to only a single MAC address:

- ATM SPA on the Cisco 7600 SIP-200

This restriction occurs because the Cisco 7600 series router keeps only one MAC address attached to an RBE PVC. The MAC address-to-PVC mapping is refreshed when a packet is received from the host. If there are multiple hosts connected to the PVC, the mapping is not stable and traffic forwarding is affected.

The solution to this problem is as follows:

1. Configure the ATM PVC for RFC 1483 bridging using the **bridge domain** *vlan x* command line interface.
2. Configure an **interface vlan** *vlan x* with the IP address of the RBE subinterface.

ATM Routed Bridge Encapsulation Configuration Task

To configure ATM routed bridge encapsulation, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot*/*subslot*/*port.subinterface* **point-to-point**	Creates the specified multipoint subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode.
Step 2	Router(config-subif)# **atm route-bridge ip**	Enables ATM RFC 1483 half-bridging (RBE bridging). **Note** The **atm route-bridge ip** command can be issued either before or after you create the PVC.
Step 3	Router(config-subif)# **ip address** *address mask* [**secondary**]	Assigns the specified IP address and subnet mask to this subinterface. This IP address should be on the same subnet as the remote bridged network (the Ethernet network).
Step 4	Router(config-subif)# **pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi*/*vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the PVC to use ILMI encapsulation (default). • **qsaal**—(Optional) Configures the PVC to use QSAAL encapsulation.
	Note When using the **pvc** command, remember that the *vpi*/*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi*/*vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.	
Step 5	Router(config-if-atm-vc)# **encapsulation aal5snap**	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type. The only supported encapsulation for an RBE PVC is **aal5snap**.
Step 6	Router(config-if-atm-vc)# **end**	Exits ATM VC configuration mode and returns to privileged EXEC mode.

Note The **atm route-bridge ip** command, like other subinterface configuration commands, is not automatically removed when you delete a subinterface. If you want to remove a subinterface and re-create it without the half-bridging, be sure to manually remove the half-bridging configuration, using the **no atm route-bridge ip** command.

Verifying the ATM Routed Bridge Encapsulation Configuration

To verify the RBE bridging configuration, use the **show ip cache verbose** command:

```
Router# show ip cache verbose

IP routing cache 3 entries, 572 bytes
   9 adds, 6 invalidates, 0 refcounts
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
   quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 00:30:34 ago

Prefix/Length       Age       Interface         Next Hop
10.1.0.51/32-24     00:30:10  Ethernet3/1/0     10.1.0.51    14
0001C9F2A81D00600939BB550800
10.8.100.50/32-24   00:00:04  ATM1/1/0.2        10.8.100.50  28
00010000AA030080C2000700000007144F5D201C0800
10.8.101.35/32-24   00:06:09  ATM1/1/0.4        10.8.101.35  28
00020000AA030080C20007000000E01E8D3F901C0800
```

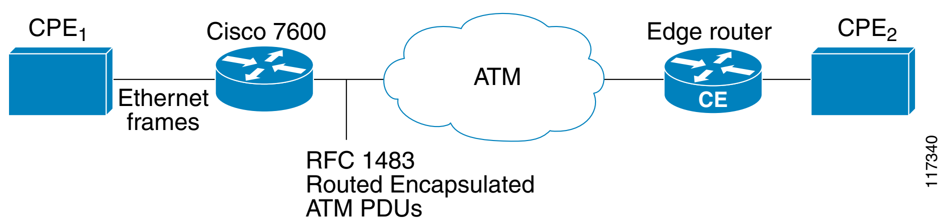
Configuring RFC 1483 Bridging of Routed Encapsulations

Bridging of routed encapsulations (BRE) enables the ATM SPA to receive RFC 1483 routed encapsulated packets and forward them as Layer 2 frames. In a BRE configuration, the PVC receives the routed PDUs, removes the RFC 1483 routed encapsulation header, and adds an Ethernet MAC header to the packet. The Layer 2 encapsulated packet is then switched by the forwarding engine to the Layer 2 interface determined by the VLAN number and destination MAC.

Note The 1-Port OC-48c/STM-16 ATM SPA does not support bridging.

Figure 7-3 shows a topology where an interface on an ATM SPA receives routed PDUs from the ATM cloud and encapsulates them as Layer 2 frames. It then forwards the frames to the Layer 2 customer device.

Figure 7-3 Example BRE Topology



RFC 1483 Bridging of Routed Encapsulations Configuration Guidelines

When configuring RFC 1483 bridging of routed encapsulations, consider the following guidelines:

- BRE requires that the ATM SPAs are installed in a Cisco 7600 SIP-200.
- PVCs must use AAL5 encapsulation.
- RFC 1483 bridged PVCs must terminate on the ATM SPA, and the traffic forwarded over this bridged connection to the edge must be forwarded through an Ethernet port.
- To use the Virtual Trunking Protocol (VTP), each main interface should have a subinterface that has been configured for the management VLANs (VLAN 1 and VLANs 1002 to 1005).
- BRE is not supported when using a Cisco 7600 SIP-400.
- Concurrent configuration of RFC 1483 bridging and BRE on the same PVC and VLAN is not supported.
- Bridging between RFC 1483 bridged PVCs is not supported.
- RFC 1483 bridging in a switched virtual circuit (SVC) environment is not supported.

RFC 1483 Bridging of Routed Encapsulations Configuration Task

To configure RFC 1483 bridging of routed encapsulations, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot*/*subslot*/*port*	Enters interface configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-if)# **no ip address**	Assigns no IP address to the interface.
Step 3	Router(config-if)# **spanning-tree bpdufilter enable**	(Optional) Blocks all Spanning Tree BPDUs on the ATM interface. This command should be used if this ATM interface is configured only for BRE VLANs. **Note** If this ATM interface is configured for both BRE and RFC 1483 bridged VLANs, do not enter this command unless you want to explicitly block BPDUs on the interface.
Step 4	Router(config-if)# **no shutdown**	Enables the interface.
Step 5	Router(config-if)# **interface atm** *slot*/*subslot*/*port.subinterface* **point-to-point**	Creates the specified point-to-point subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode.
Step 6	Router(config-subif)# **no ip address**	Assigns no IP address to the subinterface.

	Command or Action	Purpose
Step 7	Router(config-subif)# **pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi*/*vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the PVC to use ILMI encapsulation (default). • **qsaal**—(Optional) Configures the PVC to use QSAAL encapsulation.
	Note When using the **pvc** command, remember that the *vpi*/*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi*/*vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.	
Step 8	Router(config-if-atm-vc)# **bre-connect** *vlan-id* [**mac** *mac-address*]	Enables BRE bridging on the PVC, where: • **mac** *mac-address*—(Optional) Specifies the hardware (MAC) address of the destination customer premises equipment (CPE) device at the remote end of the VLAN connection.
Step 9	Router(config-if-atm-vc)# **interface gigabitethernet** *slot*/*port*	Enters interface configuration mode for the specified Gigabit Ethernet interface.
Step 10	Router(config-if)# **switchport**	Configures the Gigabit Ethernet interface for Layer 2 switching.
Step 11	Router(config-if)# **switchport access vlan** *vlan-id*	(Optional) Specifies the default VLAN for the interface. This should be the same VLAN ID that was specified in the **bre-connect** command in Step 8.
Step 12	Router(config-if)# **switchport mode access**	Puts the interface into nontrunking mode.
Step 13	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying the RFC 1483 Bridging of Routed Encapsulations Configuration

Use the following commands to verify the RFC 1483 bridging of routed encapsulations configuration:

```
Router# show running-config interface atm
10/0/3.111 Building configuration...

Current configuration : 149 bytes
!
interface ATM10/0/3.111 point-to-point  no atm enable-ilmi-trap  no
snmp trap link-status  pvc 11/101
bre-connect 11 mac 0100.1234.1234
```

```
Router# show running-config interface gigabitethernet 1/2

interface GigabitEthernet1/2
 no ip address
 switchport
 switchport access vlan 100
 no cdp enable
!

Router# show vlan id 100

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
100  VLAN0100                         active    Gi1/2, AT5/0/2

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
100  enet  100100     1500  -      -      -        -    -        0      0

Router# show atm vlan

Interface       Bridge VCD          Vlan ID
ATM4/5/0/2.1    1                   100
```

Configuring MPLS over RBE

The ATM SPAs support MLPS over RBE on a Cisco 7600 SIP-200. For more information on routed bridged encapsulation (RBE), see the "Configuring ATM Routed Bridge Encapsulation" section on page 7-21. To use this feature, configure both RBE and MPLS on the ATM subinterface using the following procedure:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-if)# **ip address**	Assigns an IP address to the interface.
Step 3	Router(config-if)# **atm route-bridge ip**	Configures RBE.
Step 4	Router(config-if)# **mpls ip**	Configures MPLS.

Verifying MPLS over RBE Configuration

Use the following commands to verify MPLS over RBE configuration:

```
Router# show running interfaces a4/1/0.200
interface ATM4/1/0.200 point-to-point
 ip address 3.0.0.2 255.255.0.0
 atm route-bridged ip
 tag-switching ip
 pvc 10/200
 !
Router# show mpls interfaces
Interface              IP            Tunnel   Operational
ATM4/1/0.200           Yes (ldp)     No       Yes

Router# show mpls ldp bindings
  tib entry: 5.0.0.0/16, rev 2
```

```
        local binding:  tag: imp-null
  tib entry: 6.0.0.0/16, rev 4
        local binding:  tag: imp-null
        remote binding: tsr: 3.0.0.1:0, tag: imp-null

Router# show mpls ldp neighbor
    Peer LDP Ident: 3.0.0.1:0; Local LDP Ident 3.0.0.2:0
        TCP connection: 3.0.0.1.646 - 3.0.0.2.11001
        State: Oper; Msgs sent/rcvd: 134/131; Downstream
        Up time: 01:51:08
        LDP discovery sources:
          ATM4/1/0.200, Src IP addr: 6.0.0.1
        Addresses bound to peer LDP Ident:
          6.0.0.1

Router# show mpls forwarding
Local  Outgoing    Prefix             Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id       switched   interface
16     Pop tag     3.0.0.0/16         0          AT4/1/0.200 6.0.0.1
17     Pop tag     16.16.16.16/32     0          AT4/1/0.200 6.0.0.1

18     19          13.13.13.13/32     134        AT4/1/0.200 6.0.0.1       <<<<<

19     Pop tag     17.17.17.17/32     0          PO8/0/0.1  point2point
```

Configuring Aggregate WRED for PVCs

Weighted Random Early Detection (WRED) is the Cisco implementation of Random Early Detection (RED) for standard Cisco IOS platforms. RED is a congestion-avoidance technique that takes advantage of the congestion-control mechanism of TCP to anticipate and avoid congestion before it occurs. By dropping packets prior to periods of high congestion, RED tells the packet source (usually TCP) to decrease its transmission rate. When configured, WRED can selectively discard lower priority traffic and provide differentiated performance characteristics for different classes of service.

The Aggregate WRED feature provides a means to overcome limitations of WRED implementations that can only support a limited number of unique subclasses. When an interface enables support for aggregate WRED, subclasses that share the same minimum threshold, maximum threshold, and mark probability values can be configured into one aggregate subclass based on their IP precedence value or differentiated services code point (DSCP) value. (The DSCP value is the first six bits of the IP type of service [ToS] byte.) You can also define a default aggregate subclass for all subclasses that have not been explicitly defined.

For more complete information on WRED, refer to the *Cisco IOS Quality of Service Solutions Configuration Guide*.

Aggregate WRED Configuration Guidelines

When configuring aggregate WRED on an ATM SPA interface, consider the following guidelines:

- The Aggregate WRED feature requires that the ATM SPAs are installed in a Cisco 7600 SIP-200 or a Cisco 7600 SIP-400.
- With the Aggregate WRED feature, the previous configuration limitation of a maximum of 6 precedence values per class per WRED policy map is no longer in effect.

- When you configure a policy map class for aggregated WRED on an ATM interface, then you cannot also configure the standard **random-detect** commands in interface configuration or policy-map class configuration mode.
- Specifying the **precedence-based** keyword is optional, **precedence-based** is the default form of aggregate WRED.
- The set of subclass values (IP precedence or DSCP) defined on a **random-detect precedence (aggregate)** or **random-detect dscp (aggregate)** CLI will be aggregated into a single hardware WRED resource. The statistics for these subclasses will also be aggregated.
- Defining WRED parameter values for the default aggregate class is optional. If defined, WRED parameters applied to the default aggregate class will be used for all subclasses that have not been explicitly configured. If all possible IP precedence or DSCP values are defined as subclasses, a default specification is unnecessary. If the optional parameters for a default aggregate class are not defined and packets with an unconfigured IP precedence or DSCP value arrive at the interface, these undefined subclass values will be set based on interface (VC) bandwidth.
- After aggregate WRED has been configured in a service policy map, the service policy map must be applied at the ATM VC level (as shown in Step 5 through Step 8 of "Configuring Aggregate WRED Based on IP Precedence").
- The Aggregate WRED feature is not supported in a switched virtual circuit (SVC) environment.

Configuring Aggregate WRED Based on IP Precedence

To configure aggregate WRED to drop packets based on IP precedence values, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **policy-map** *policy-map-name*	Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy. • *policy-map-name*—Name of a service policy map to be created. The name can be a maximum of 40 alphanumeric characters.
Step 2	Router(config-pmap)# **class** {*class-name* | **class-default**}	Specifies the class policy to be configured. • *class-name*—Name of class you want to configure. Note that WRED can be defined for a user-defined class only if the class has the bandwidth/shape feature enabled. • **class-default**—Default class.

	Command	Purpose
Step 3	Router(config-pmap-c)# **random-detect** [**precedence-based**] **aggregate** [**minimum-thresh** *min-thresh* **maximum-thresh** *max-thresh* **mark-probability** *mark-prob*]	Enables aggregate WRED based on IP precedence values. If optional parameters for a default aggregate class are not defined, these parameters will be set based on interface (VC) bandwidth. • **precedence-based**—(Optional) Specifies that aggregate WRED is to drop packets based on IP precedence values. This is the default. • *min-thresh*—(Optional) Minimum threshold in number of packets. The value range of this argument is from 1 to 12288. • *max-thresh*—(Optional) Maximum threshold in number of packets. The value range of this argument is from the value of the minimum threshold argument to 12288. • *mark-prob*—(Optional) Denominator for the fraction of packets dropped when the average queue depth is at the maximum threshold. The value range is from 1 to 255.
Step 4	Router(config-pmap-c)# **random-detect precedence values** *sub-class-val1* [...[*sub-class-val8*]] **minimum-thresh** *min-thresh* **maximum-thresh** *max-thresh* [**mark-probability** *mark-prob*]	Configures the WRED parameters for packets with one or more specific IP precedence values. • *sub-class-val1* [...[*sub-class-val8*]]—One or more specific IP precedence values to which the following WRED profile parameter specifications are to apply. A maximum of 8 subclasses (IP precedence values) can be specified per CLI entry. The IP precedence value can be a number from 0 to 7. • *min-thresh*—Minimum threshold in number of packets. The value range of this argument is from 1 to 12288. • *max-thresh*—Maximum threshold in number of packets. The value range of this argument is from the value of the minimum threshold argument to 12288. • *mark-prob*—Denominator for the fraction of packets dropped when the average queue depth is at the maximum threshold. The value range is from 1 to 255. Repeat this command for each set of IP precedence values that share WRED parameters.

	Command	Purpose
Step 5	Router(config-pmap-c)# **interface atm** *slot*/*subslot*/*port*.*subinterface* **point-to-point**	Creates the specified point-to-point subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode. • *slot*—Chassis slot number where the SIP is installed. • *subslot*—Secondary slot of the SIP where the SPA is installed. • *port* —Number of the individual interface port on the SPA. • .*subinterface*—Subinterface number. The number that precedes the period must match the number to which this subinterface belongs. The range is 1 to 4,294,967,293.
Step 6	Router(config-subif)# **ip address** *address mask*	Assigns the specified IP address and subnet mask to the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 7	Router(config-subif)# **pvc** [*name*] *vpi*/*vci* [*ilmi* | *qsaal*]	Configures a new ATM PVC by assigning an optional name and its VPI/VCI numbers. • *name*—(Optional) An arbitrary string that identifies this PVC. • *vpi*—VPI ID. The range is 0 to 255. • *vci*—VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except 5 for the QSAAL PVC and 16 for the ILMI PVC.
Step 8	Router(config-subif)# **service-policy output** *policy-map-name*	Attaches the specified policy map to the subinterface. • *policy-map-name*—Name of a service policy map to be attached. The name can be a maximum of 40 alphanumeric characters.

Verifying the Precedence-Based Aggregate WRED Configuration

To verify a precedence-based aggregate WRED configuration, use the **show policy-map interface** command. Note that the statistics for IP precedence values 0 through 3 and 4 and 5 have been aggregated into one line each.

```
Router# show policy-map interface a4/1/0.10
 ATM4/1/0.10: VC 10/110 -

 Service-policy output: prec-aggr-wred

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
        Exp-weight-constant: 9 (1/512)
```

```
Mean queue depth: 0
class       Transmitted      Random drop       Tail drop      Minimum   Maximum  Mark
              pkts/bytes       pkts/bytes       pkts/bytes    thresh  thresh  prob

0  1  2  3        0/0              0/0              0/0            10      100  1/10
4  5              0/0              0/0              0/0            40      400  1/10
6                 0/0              0/0              0/0            60      600  1/10
7                 0/0              0/0              0/0            70      700  1/10
```

Configuring Aggregate WRED Based on DSCP

To configure aggregate WRED to drop packets based on the differentiated services code point (DSCP) value, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **policy-map** *policy-map-name*	Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy. • *policy-map-name*—Name of a service policy map to be created. The name can be a maximum of 40 alphanumeric characters.
Step 2	Router(config-pmap)# **class** {*class-name* | **class-default**}	Specifies the class policy to be configured. • *class-name*—Name of class you want to configure. Note that WRED can be defined for a user-defined class only if the class has the bandwidth/shape feature enabled. • **class-default**—Default class.
Step 3	Router(config-pmap-c)# **random-detect dscp-based aggregate** [**minimum-thresh** *min-thresh* **maximum-thresh** *max-thresh* **mark-probability** *mark-prob*]	Enables aggregate WRED based on DSCP values. If optional parameters for a default aggregate class are not defined, these parameters will be set based on interface (VC) bandwidth. • *min-thresh*—(Optional) Minimum threshold in number of packets. The value range of this argument is from 1 to 12288. • *max-thresh*—(Optional) Maximum threshold in number of packets. The value range of this argument is from the value of the minimum threshold argument to 12288. • *mark-prob*—(Optional) Denominator for the fraction of packets dropped when the average queue depth is at the maximum threshold. The value range is from 1 to 255.

	Command	Purpose
Step 4	Router(config-pmap-c)# **random-detect dscp values** *sub-class-val1* [...[*sub-class-val8*]] **minimum-thresh** *min-thresh* **maximum-thresh** *max-thresh* [**mark-probability** *mark-prob*]	Configures the WRED parameters for packets with one or more specific DSCP values. • *sub-class-val1* [...[*sub-class-val8*]]—One or more DSCP values to which the following WRED parameter specifications are to apply. [A maximum of 8 subclasses (IP precedence values) can be specified per CLI entry.] The DSCP value can be a number from 0 to 63, or it can be one of the following keywords: ef, af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, cs1, cs2, cs3, cs4, cs5, or cs7 • *min-thresh*—Specifies the minimum threshold in number of packets. The value range of this argument is from 1 to 12288. • *max-thresh*—Specifies the maximum threshold in number of packets. The value range of this argument is from the value of the minimum threshold argument to 12288. • *mark-prob*—Specifies the denominator for the fraction of packets dropped when the average queue depth is at the maximum threshold. The value range is from 1 to 255. Repeat this command for each set of DSCP values that share WRED parameters.
Step 5	Router(config-pmap-c)# **interface atm** *slot/subslot/port.subinterface* **point-to-point**	Creates the specified point-to-point subinterface on the given port on the specified ATM SPA, and enters subinterface configuration mode. • *slot*—Chassis slot number where the SIP is installed. • *subslot*—Secondary slot of the SIP where the SPA is installed. • *port*—Number of the individual interface port on the SPA. • *.subinterface*—subinterface number. The number that precedes the period must match the number to which this subinterface belongs. The range is 1 to 4,294,967,293.
Step 6	Router(config-subif)# **ip address** *address mask*	Assigns the specified IP address and subnet mask to the interface. • *address*—IP address. • *mask*—Subnet mask.

	Command	Purpose
Step 7	Router(config-subif)# **pvc** [*name*] *vpi*/*vci* [*ilmi* | *qsaal*]	Configures a new ATM PVC by assigning an optional name and its VPI/VCI numbers. • *name*—(Optional) An arbitrary string that identifies this PVC. • *vpi*—VPI ID. The range is 0 to 255. • *vci*—VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except 5 for the QSAAL PVC and 16 for the ILMI PVC.
Step 8	Router(config-subif)# **service-policy output** *policy-map-name*	Attaches the specified policy map to the subinterface. • *policy-map-name*—Name of a service policy map to be attached. The name can be a maximum of 40 alphanumeric characters

Verifying the DSCP-Based Aggregate WRED Configuration

To verify a DSCP-based aggregate WRED configuration, use the **show policy-map interface** command. Note that the statistics for DSCP values 0 through 3, 4 through 7, and 8 through 11 have been aggregated into one line each.

```
Router# show policy-map interface a4/1/0.11
 ATM4/1/0.11: VC 11/101 -

  Service-policy output: dscp-aggr-wred

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
        Exp-weight-constant: 0 (1/1)
        Mean queue depth: 0
        class       Transmitted     Random drop      Tail drop     Minimum   Maximum  Mark
                  pkts/bytes       pkts/bytes       pkts/bytes    thresh  thresh  prob
        default           0/0              0/0              0/0             1       10  1/10
        0  1  2  3
        4  5  6  7        0/0              0/0              0/0            10       20  1/10
        8  9  10 11       0/0              0/0              0/0            10       40  1/10
```

Creating and Configuring Switched Virtual Circuits

A switched virtual circuit (SVC) is created and released dynamically, providing user bandwidth on demand. To enable the use of SVCs, you must configure a signaling protocol to be used between the Cisco 7600 series router and the ATM switch. The ATM SPA supports versions 3.0, 3.1, and 4.0 of the User-Network Interface (UNI) signaling protocol, which uses the Integrated Local Management Interface (ILMI) to establish, maintain, and clear the ATM connections at the UNI.

The Cisco 7600 series router does not perform ATM-level call routing when configured for UNI/ILMI operation. Instead, the ATM switch acts as the network and performs the call routing, while the Cisco 7600 series router acts only as the user end-point of the call circuit and only routes packets through the resulting circuit.

Note The 1-Port OC-48c/STM-16 ATM SPA does not support SVCs.

To use UNI/ILMI signaling, you must create an ILMI PVC and a signaling PVC to be used for the SVC call-establishment and call-termination messages between the ATM switch and Cisco 7600 series router. This also requires configuring the ATM interface with a network service access point (NSAP) address that uniquely identifies itself across the network.

The NSAP address consists of a network prefix (13 hexadecimal digits), a unique end station identifier (ESI) of 6 hexadecimal bytes, and a selector byte. If an ILMI PVC exists, the Cisco 7600 series router can obtain the NSAP prefix from the ATM switch, and you must manually configure only the ESI and selector byte. If an ILMI PVC does not exist, or if the ATM switch does not support this feature, you must configure the entire address manually.

To create and configure an SVC, use the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot*/*subslot*/*port*	Enters interface configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-subif)# **pvc** [*name*] **0/5 qsaal**	Configures a new ATM PVC to be used for SVC signaling: • *name*—(Optional) An arbitrary string that identifies this PVC. • *vpi*—Specifies the VPI ID. The valid range is 0 to 255, but the recommended value for *vpi* for the signaling PVC is 0. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535, but the recommended value for *vci* for the QSAAL signaling PVC is 5. **Note** The ATM switch must be configured with the same VPI and VCI values for this PVC. • **qsaal**—Configures the signaling PVC to use QSAAL encapsulation.

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td>Step 3</td><td>Router(config-subif)# pvc [name] 0/16 ilmi</td><td>Creates a new ATM PVC to be used for ILMI signaling:
• name—(Optional) An arbitrary string to identify the PVC.
• vpi—Specifies the VPI ID. The valid range is 0 to 255, but the recommended value for vpi for the ILMI PVC is 0.
• vci—Specifies the VCI ID. The valid range is 1 to 65535, but the recommended value for vci for the ILMI PVC is 16.
• ilmi—Configures the PVC to use ILMI encapsulation.</td></tr>
<tr><td></td><td colspan="2">Note The signaling and ILMI PVCs must be set up on the main ATM interface, not on a subinterface.</td></tr>
<tr><td>Step 4</td><td>Router(config-if-atm-vc)# exit</td><td>Exits ATM PVC configuration mode and returns to interface configuration mode.</td></tr>
<tr><td>Step 5</td><td>Router(config-if)# atm ilmi-keepalive [seconds] [retry counts]</td><td>(Optional) Enables ILMI keepalive messages and sets the interval between them. ILMI keepalive messages are disabled by default.
• seconds—(Optional) The amount of time, in seconds, between keepalive messages between the Cisco 7600 series router and the ATM switch. The valid range is 1 to 65535, with a default of 3 seconds.
• retry counts—(Optional) Specifies the number of times the router should resend a keepalive message if the first message is unacknowledged. The valid range is 2 to 5, with a default of 4.</td></tr>
<tr><td>Step 6</td><td>Router(config-if)# atm esi-address esi.selector

or</td><td>Specifies the end station ID (ESI) and selector fields for the local portion of the interface's NSAP address, and configures the interface to get the NSAP prefix from the ATM switch.
• esi—Specifies a string of 12 hexadecimal digits, in dotted notation, for the ATM interface's ESI value. This value must be unique across the network.
• selector—Specifies a string of 2 hexadecimal digits for the selector byte for this ATM interface.
To configure the ATM address, you need to enter only the ESI (12 hexadecimal digits) and the selector byte (2 hexadecimal digits). The NSAP prefix (26 hexadecimal digits) is provided by the ATM switch.</td></tr>
<tr><td></td><td>Router(config-if)# atm nsap-address nsap-address</td><td>Assigns a complete NSAP address (40 hexadecimal digits) to the interface. The address consists of a network prefix, ESI, and selector byte, and must be in the following format:
<code>XX.XXXX.XX.XXXXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XX</code>
Note The above dotted hexadecimal format provides some validation that the address is a legal value. If you know that the NSAP address is correct, you may omit the dots.</td></tr>
</table>

	Command or Action	Purpose
	Note The **atm esi-address** and **atm nsap-address** commands are mutually exclusive. Configuring the Cisco 7600 series router with one of these commands automatically negates the other. Use the **show interface atm** command to display the NSAP address that is assigned to the interface.	
Step 7	Router(config-if)# **interface atm** *slot/subslot/port.subinterface* [**multipoint** | **point-to-point**]	(Optional) Creates the specified subinterface on the specified ATM interface, and enters subinterface configuration mode. **Note** You can create SVCs on either the main ATM interface or on a multipoint subinterface.
Step 8	Router(config-subif)# **svc** [*name*] **nsap** *address*	Creates an SVC and specifies the destination NSAP address (40 hexadecimal digits in dotted notation). You can also configure the following option: • *name*—(Optional) An arbitrary string that identifies this SVC.
Step 9	Router(config-if-atm-vc)# **oam-svc** [**manage**] [*frequency*]	Enables end-to-end Operation, Administration, and Maintenance (OAM) loopback cell generation and management of the SVC. • **manage**—(Optional) Enables OAM management of the SVC. • *frequency*—(Optional) Specifies the delay between transmitting OAM loopback cells. The valid range is 0 to 600 seconds, with a default of 10 seconds.
Step 10	Router(config-if-atm-vc)# **protocol** *protocol* {*protocol-address* | **inarp**} [[**no**] **broadcast**]	Configures the SVC for a particular protocol and maps it to a specific *protocol-address*. • *protocol*—Typically set to either **ip** or **ppp**, but other values are possible. • *protocol-address*—Destination address or virtual interface template for this SVC (if appropriate for the *protocol*). • **inarp**—Specifies that the SVC uses Inverse ARP to determine its address. • [**no**] **broadcast**—(Optional) Specifies that this mapping should (or should not) be used for broadcast packets.
Step 11	Router(config-if-atm-vc)# **encapsulation aal5snap**	(Optional) Configures the ATM adaptation layer (AAL) and encapsulation type. The default and only supported type is **aal5snap**.
	Note Repeat Step 7 through Step 11 for each SVC to be created.	
Step 12	Router(config-if-atm-vc)# **end**	Exits SVC configuration mode and returns to privileged EXEC mode.

Verifying the SVC Configuration

Use the **show atm svc** and **show atm ilmi-status** commands to verify the configuration of the SVCs that are currently configured on the Cisco 7600 series router.

```
Router# show atm svc

           VCD /                                         Peak  Avg/Min Burst
Interface  Name         VPI   VCI  Type   Encaps    SC   Kbps   Kbps   Cells  Sts
4/0/0      1              0     5  SVC    SAAL      UBR  155000                UP
4/0/2      4              0    35  SVC    SNAP      UBR  155000                UP
4/1/0      16             0    47  SVC    SNAP      UBR  155000                UP
4/1/0.1    593            0    80  SVC    SNAP      UBR  155000            UP
```

Tip To display all SVCs on a particular ATM interface or subinterface, use the **show atm svc interface atm** command.

To display detailed information about a particular SVC, specify its VPI and VCI values:

```
Router# show atm svc 0/35

ATM5/1/0.200: VCD: 3384, VPI: 0, VCI: 35, Connection Name: SVC00
UBR, PeakRate: 155000
AAL5-MUX, etype:0x800, Flags: 0x44, VCmode: 0x0
OAM frequency: 10 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Received
OAM VC status: Verified
ILMI VC status: Not Managed
VC is managed by OAM.
InARP DISABLED
Transmit priority 6
InPkts: 0, OutPkts: 4, InBytes: 0, OutBytes: 400
InPRoc: 0, OutPRoc: 4, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0,  OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0, LengthViolation: 0, CPIErrors: 0
Out CLP=1 Pkts: 0
OAM cells received: 10
F5 InEndloop: 10, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
OAM cells sent: 10
F5 OutEndloop: 10, F5 OutSegloop: 0, F5 OutRDI: 0
F4 OutEndloop: 0, F4 OutSegloop: 0, F4 OutRDI: 0
OAM cell drops: 0
Status: UP
TTL: 4
interface =  ATM5/1/0.200, call locally initiated, call reference = 8094273
vcnum = 3384, vpi = 0, vci = 35, state = Active(U10)
, point-to-point call
Retry count: Current = 0
timer currently inactive, timer value = 00:00:00
Remote Atm Nsap address: 47.00918100000000107B2B4B01.111155550001.00
, VC owner: ATM_OWNER_SMAP
```

To display information about the ILMI status and NSAP addresses being used for the SVCs on an ATM interface, use the **show atm ilmi-status** command:

```
Router# show atm ilmi-status atm 4/1/0

Interface : ATM4/1/0 Interface Type : Private UNI (User-side)
ILMI VCC : (0, 16) ILMI Keepalive : Enabled/Up (5 Sec 4 Retries)
ILMI State:       UpAndNormal
Peer IP Addr:     10.10.13.1       Peer IF Name:     ATM 3/0/3
Peer MaxVPIbits:  8               Peer MaxVCIbits:  14
Active Prefix(s) :
47.0091.8100.0000.0010.11b8.c601
End-System Registered Address(s) :
47.0091.8100.0000.0010.11b8.c601.2222.2222.2222.22(Confirmed)
47.0091.8100.0000.0010.11b8.c601.aaaa.aaaa.aaaa.aa(Confirmed)
```

Tip To display information about the SVC signaling PVC and ILMI PVC, use the **show atm pvc 0/5** and **show atm pvc 0/16** commands.

Configuring Traffic Parameters for PVCs or SVCs

After creating a PVC or SVC, you can also configure it for the type of traffic quality of service (QoS) class to be used over the circuit:

- Constant Bit Rate (CBR)—Configures the CBR service class and specifies the average cell rate for the PVC or SVC.
- Unspecified Bit Rate (UBR)—Configures the UBR service class and specifies the output peak rate (PCR) for the PVC or SVC. This is the default configuration. SVCs can also be configured with similar input parameters.
- Unspecified Bit Rate Plus (UBR+)—Configures the UBR+ service class and specifies the output peak cell rate (PCR) and minimum cell rate (MCR) for the SVC. SVCs can also be configured with similar input parameters.

Note The 1-Port OC-48c/STM-16 ATM SPA does not support UBR+.

- Variable Bit Rate–Non-real Time (VBR-nrt)—Configures the VBR-nrt service class and specifies the output PCR, output sustainable cell rate (SCR), and output maximum burst size (MBS) for the PVC or SVC. SVCs can also be configured with similar input parameters.
- Variable Bit Rate–Real Time (VBR-rt)—Configures the VBR-rt service class and the peak rate and average rate burst for the PVC or SVC.

Each service class is assigned a different transmit priority, which the Cisco 7600 series router uses to determine which queued cell is chosen to be transmitted out of an interface during any particular cell time slot. This ensures that real-time QoS classes have a higher likelihood of being transmitted during periods of congestion. Table 7-1 lists the ATM QoS classes and their default transmit priorities.

Table 7-1 ATM Classes of Service and Default Transmit Priorities

Service Category	Transmit Priority[1]
Signaling, Operation, Administration, and Maintenance (OAM) cells, and other control cells	0 (highest)
CBR when greater than 5 percent of the line rate	1
CBR when less than 5 percent of the line rate	2
Voice traffic	3
VBR-rt	4
VBR-nrt	5
UBR	6
Unused and not available or configurable	7 (lowest)

1. The default priorities can be changed for individual VCs using the **transmit-priority** VC configuration command.

Note When using a CBR VC that exceeds half of the interface line rate, it is possible in some cases that the shaping accuracy for the CBR traffic can drop from 99 percent to 98 percent when the interface is also configured for UBR VCs that are oversubscribed (that is, the UBR VCs are configured for a total line rate that exceeds the interface line rate). If this small drop in accuracy is not acceptable, then we recommend using VBR-rt or VBR-nrt instead of CBR when oversubscribing UBR traffic.

You can configure a PVC or SVC for only one QoS service class. If you enter more than one type, only the most recently configured QoS class takes effect on the circuit.

To configure the traffic parameters for a PVC or SVC, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot*/*subslot* or Router(config)# **interface atm** *slot*/*subslot*/*port*.*subinterface* [**multipoint** | **point-to-point**]	Enters interface or subinterface configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-if)# **pvc** [*name*] *vpi*/*vci* or Router(config-if)# **svc** [*name*] *nsap-address*	Specifies the PVC or SVC to be configured, and enters PVC/SVC configuration mode.

Note When using the **pvc** command, remember that the *vpi*/*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi*/*vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.

	Command or Action	Purpose
Step 3	Router(config-if-atm-vc)# **cbr** *rate*	Configures constant bit rate (CBR) quality of service (QoS) and average cell rate for the PVC or SVC: • *rate*—Average cell rate in kbps. The valid range is 48 to 149760 (OC-3) or 599040 (OC-12).
	or	
	Router(config-if-atm-vc)# **ubr** *output-pcr* [*input-pcr*]	Configures unspecified bit rate (UBR) quality of service (QoS) and peak cell rate (PCR) for the PVC or SVC: • *output-pcr*—Output PCR in kbps. The valid range is 48 to 149760 (OC-3), 599040 (OC-12), or 2396160 (1-Port OC-48c/STM-16 ATM SPA). • *input-pcr*—(Optional for SVCs only) Input PCR in kbps. If omitted, *input-pcr* equals *output-pcr*.
	or	
	Router(config-if-atm-vc)# **vbr-nrt** *output-pcr output-scr output-mbs* [*input-pcr*] [*input-scr*] [*input-mbs*]	Configures the variable bit rate–nonreal time (VBR-nrt) QoS, the peak cell rate (PCR), sustainable cell rate (SCR), and maximum burst cell size (MBS) for the PVC or SVC: • *output-pcr*—Output PCR in kbps. The valid range is 48 to 149760 (OC-3), 599040 (OC-12), or 2396160 (1-Port OC-48c/STM-16 ATM SPA). • *output-scr*—Output SCR in kbps. The valid range is 48 to PCR, and typically is less than the PCR value. • *output-mbs*—Output MBS in number of cells. The valid range is 1 to 65535, depending on the PCR and SCR values. If the PCR and SCR are configured to the same value, the only valid value for MBS is 1. • *input-pcr*—(Optional for SVCs only) Input PCR in kbps. • *input-scr*—(Optional for SVCs only) Input SCR in kbps. • *input-mbs*—(Optional for SVCs only) Input MBS in number of cells.
	or	
	Router(config-if-atm-vc)# **vbr-rt** *pcr scr burst*	Configures the variable bit rate–real time (VBR-rt) QoS, and the PCR, average cell rate (ACR), and burst cell size (BCS) for the PVC or SVC: • *pcr*—PCR in kbps. The valid range is 48 to 149760 (OC-3), 599040 (OC-12), or 2396160 (1-Port OC-48c/STM-16 ATM SPA). • *scr*—SCR in kbps. The valid range is 48 to PCR, and typically is less than the PCR value. • *burst*—Burst size in number of cells. The valid range is 1 to 65535, depending on the PCR and SCR values. If the PCR and SCR are configured to the same value, the only valid value for *burst* is 1.

	Command or Action	Purpose
Step 4	Router(config-if-atm-vc)# **transmit-priority** *level*	(Optional) Configures the PVC for a new transmit priority level. • *level*—Priority level from 1 to 6. The default value is determined by the PVC's configured service class (see Table 7-1 on page 7-39 for the default levels).
	Note Repeat Step 2 through Step 4 for each PVC or SVC to be configured.	
Step 5	Router(config-if-atm-vc)# **end**	Exits PVC/SVC configuration mode and returns to privileged EXEC mode.

Verifying the Traffic Parameter Configuration

Use the **show atm vc** command to verify the configuration of the traffic parameters for a PVC or SVC:

```
Router# show atm vc 20

   ATM1/1/0.200: VCD: 20, VPI: 2, VCI: 200
   UBR, PeakRate: 44209
   AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
   OAM frequency: 0 second(s)
   InARP frequency: 5 minutes(s)
   Transmit priority 4
   InPkts: 10, OutPkts: 11, InBytes: 680, OutBytes: 708
   InPRoc: 10, OutPRoc: 5, Broadcasts: 0
   InFast: 0, OutFast: 0, InAS: 0, OutAS: 6
   InPktDrops: 0, OutPktDrops: 0
   CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0
   OAM cells received: 0
   OAM cells sent: 0
   Status: UP
```

To verify the configuration of all PVCs or SVCs on an interface, use the **show atm vc interface atm** command:

```
Router# show atm vc interface atm 2/1/0

ATM2/1/0.101: VCD: 201, VPI: 20, VCI: 101
UBR, PeakRate: 149760
AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 0 second(s)
InARP frequency: 15 minutes(s)
Transmit priority 4
InPkts: 3153520, OutPkts: 277787, InBytes: 402748610, OutBytes: 191349235
InPRoc: 0, OutPRoc: 0, Broadcasts: 0
InFast: 211151, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0, OutPktDrops: 17
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0
OAM cells received: 0
OAM cells sent: 0
Status: UP
```

Configuring Virtual Circuit Classes

When multiple PVCs or SVCs use the same or similar configurations, you can simplify the Cisco 7600 series router's configuration file by creating virtual circuit (VC) classes. Each VC class acts as a template, which you can apply to an ATM interface or subinterface, or to individual PVCs or SVCs.

When you apply a VC class to an ATM interface or subinterface, all PVCs and SVCs created on that interface or subinterface inherit the VC class configuration. When you apply a VC class to an individual PVC or SVC, that particular PVC or SVC inherits the class configuration.

You can then customize individual PVCs and SVCs with further configuration commands. Any commands that you apply to individual PVCs and SVCs take precedence over those of the VC class that were applied to the interface or to the PVC/SVC.

To create and configure a VC class, and then apply it to an interface, subinterface, or individual PVC or SVC, use the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **vc-class atm** *vc-class-name*	Creates an ATM virtual circuit (VC) class and enters VC-class configuration mode. • *vc-class-name*—Arbitrary name to identify this particular VC class.
Step 2	Router(config-vc-class)# *configuration-commands*	Enter any PVC or SVC configuration commands for this VC class. See the "Creating a Permanent Virtual Circuit" section on page 7-7 and the "Creating and Configuring Switched Virtual Circuits" section on page 7-34 for additional information. **Note** You can specify both PVC and SVC configuration commands in the same VC class. If a command is not appropriate for a PVC or SVC, it is ignored when the VC class is assigned to the PVC or SVC.
Step 3	Router(config-vc-class)# **interface atm** *slot/subslot/port* or Router(config-vc-class)# **interface atm** *slot/subslot/port.subinterface* [**multipoint** | **point-to-point**]	Enters subinterface configuration mode for the specified ATM interface or subinterface.
Step 4	Router(config-if)# **class-int** *vc-class-name*	(Optional) Applies a VC class on the ATM main interface or subinterface. This class then applies to all PVCs or SVCs that are created on that interface. • *vc-class-name*—Name of the VC class that was created in Step 1.
Step 5	Router(config-if)# **pvc** [*name*] *vpi/vci* or Router(config-if)# **svc** [*name*] *nsap-address*	Specifies the PVC or SVC to be configured, and enters ATM VC configuration mode.

Note When using the **pvc** command, remember that the *vpi/vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi/vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.

	Command or Action	Purpose
Step 6	Router(config-if-atm-vc)# **class-vc** *vc-class-name*	Assigns the specified VC class to this PVC or SVC. • *vc-class-name*—Name of the VC class that was created in Step 1.
Step 7	Router(config-if-atm-vc)# *configuration-commands*	Any other VC configuration commands to be applied to this particular PVC or SVC. Commands that are applied to the individual PVC or SVC supersede any conflicting commands that were specified in the VC class.
Step 8	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying the Virtual Circuit Class Configuration

To verify the virtual circuit class configuration, use the **show atm vc** command:

```
Router# show atm vc

            VCD /                                           Peak  Avg/Min Burst
Interface   Name         VPI   VCI  Type    Encaps     SC   Kbps    Kbps   Cells  Sts
6/1/0       1              0     5  PVC     SAAL       UBR  155000                 UP
6/1/0       2              0    16  PVC     ILMI       UBR  155000                 UP
6/1/0.1     3              1    32  PVC-D   SNAP       UBR  155000                 UP
6/1/0.2     4              2    32  PVC-D   SNAP       UBR  155000                 UP
```

Configuring Virtual Circuit Bundles

Virtual circuit bundles are similar to VC classes, in that they allow you to configure a large group of PVCs by configuring a template (the VC bundle). The main difference between a VC bundle and a VC class is that the VC bundle management allows you to configure multiple VCs that have different QoS characteristics between any pair of ATM-connected routers.

Using VC bundles, you first create an ATM VC bundle and then add VCs to it, and each VC in the bundle can have its own ATM traffic class and ATM traffic parameters. You can configure the VCs collectively at the bundle level, or you can configure the individual VC bundle members. You can also apply a VC class to a bundle to apply the VC class configuration to all of the VCs in the bundle.

You can therefore create differentiated service by mapping one or more MPLS EXP levels to each VC in the bundle, thereby enabling individual VCs in the bundle to carry packets marked with different MPLS EXP levels. The ATM VC bundle manager determines which VC to use for a particular packet by matching the MPLS EXP level of the packet to the MPLS EXP levels assigned to the VCs in the bundle. The bundle manager can also use Weighted Random Early Detection (WRED) or distributed WRED (dWRED) to further differentiate service across traffic that has different MPLS EXP levels.

Virtual Circuit Bundles Configuration Guidelines

- VC bundles are supported only on ATM SPAs in a Cisco 7600 SIP-200. Bundles are not supported for ATM SPAs in a Cisco 7600 SIP-400.
- VC bundles can be used only for PVCs, not SVCs.
- VC bundles require ATM PVC management, as well as Forwarding Information Base (FIB) and Tag Forwarding Information Base (TFIB) switching functionality.
- The Cisco 7600 series router at the remote end of the network must be using a version of Cisco IOS that supports MPLS and ATM PVC management.

Virtual Circuit Bundles Configuration Task

To create and configure a VC bundle and then apply it to an ATM interface or subinterface, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **ip cef** [**distributed**]	Enables Cisco Express Forwarding (CEF) Layer 3 switching on the Cisco 7600 series router. The Cisco 7600 series router enables CEF by default. • **distributed**—(Optional) Enables distributed CEF (dCEF).
Step 2	Router(config)# **mpls label protocol ldp**	Specifies the default label distribution protocol for a platform.
Step 3	Router(config)# **interface atm** *slot/subslot/port* or Router(config)# **interface atm** *slot/subslot/port.subinterface* [**multipoint** | **point-to-point**]	Enters interface configuration mode for the specified ATM interface or subinterface.
Step 4	Router(config-if)# **mpls ip**	Enables MPLS forwarding of IPv4 packets along normally routed paths for the interface.
Step 5	Router(config-if)# **bundle** *bundle-name*	Creates an ATM virtual circuit (VC) bundle and enters bundle configuration mode. • *bundle-name*—Arbitrary name to identify this particular VC bundle.
Step 6	Router(config-if-atm-bundle)# **class-bundle** *vc-class-name*	(Optional) Applies a VC class to this bundle. The class configuration is then applied to all VCs in the bundle. • *vc-class-name*—Name of the VC class to be applied to this bundle and its PVCs or SVCs. See the "Configuring Virtual Circuit Classes" section on page 7-42 for information on creating VC classes.
Step 7	Router(config-if-atm-bundle)# *configuration-commands*	Enter any other PVC or SVC configuration commands for this VC bundle. See the "Creating a Permanent Virtual Circuit" section on page 7-7 and the "Creating and Configuring Switched Virtual Circuits" section on page 7-34 for additional information.
	Note Configuration commands applied directly to the VC bundle supersede a configuration that is applied through a VC class.	
Step 8	Router(config-if-atm-bundle)# **pvc-bundle** [*name*] *vpi/vci*	Creates a member PVC of the bundle and enters PVC bundle configuration mode.

	Command or Action	Purpose
Step 9	Router(config-if-atm-member)# **mpls experimental** [*level* | **other** | *range*]	(Optional) Configures the MPLS EXP levels for the PVC bundle member. • *level*—MPLS EXP level for the PVC bundle member. The valid range is 0 to 7. • **other**—Any MPLS EXP levels in the range from 0 to 7 that are not explicitly configured (default). • *range*—A range of MPLS EXP levels between 0 and 7, separated by a hyphen.
Step 10	Router(config-if-atm-member)# **bump** {**implicit** | **explicit** *precedence-level* | **traffic**}	(Optional) Configures the bumping rules for the PVC bundle member. • **implicit—**Bumped traffic is carried by a VC with a lower precedence (default). • **explicit** *precedence-level*—Specifies the precedence level of the traffic that should be bumped when the PVC member goes down. The *precedence-level* can range from 0 to 9. • **traffic**—The PVC member accepts bumped traffic (default). Use **no bump traffic** to specify that the PVC member does not accept bumped traffic.
Step 11	Router(config-if-atm-member)# **protect** {**group** | **vc**}	(Optional) Specifies that the PVC bundle member is protected. • **group**—Specifies that the PVC bundle member is part of a protected group. When all members of a protected group go down, the bundle goes down. • **vc**—Specifies that the PVC bundle member is individually protected. When a protected VC goes down, it also takes the bundle down. By default, PVC bundle members are not protected.
Step 12	Router(config-if-atm-member)# *configuration-commands*	Any other VC configuration commands to be applied to this particular VC bundle member. Commands that are applied to a bundle member supersede any conflicting commands that were specified in the VC class or VC bundle.
	Note Repeat Step 8 through Step 12 for each PVC member of the bundle to be created.	
Step 13	Router(config-if-atm-member)# **end**	Exits PVC bundle configuration mode and returns to privileged EXEC mode.

Verifying the Virtual Circuit Bundles Configuration

To verify the configuration of the virtual circuit bundles and display the configuration for its interface or subinterface, use the **show running-config interface atm** command, as in the following example:

```
Router# show running-config interface atm 4/1/0.2

interface ATM4/1/0.2 point-to-point
 ip address 10.10.10.1 255.255.255.0
 no ip directed-broadcast
 no atm enable-ilmi-trap
```

```
 bundle ABC
  class-bundle bundle-class
  pvc-bundle ABC-high 1/107
   class-vc high
  pvc-bundle ABC-med 1/105
   class-vc med
  pvc-bundle ABC-low 1/102
   class-vc low
 !
!
```

To verify the operation and current status of a virtual circuit bundle, specify the bundle name with the **show atm bundle** command:

```
Router# show atm bundle ABC

ABC on ATM4/1/0.2: UP

                       Config     Current    Bumping  PG/  Peak  Avg/Min Burst
VC Name       VPI/ VCI Prec/Exp   Prec/Exp   PrecExp/ PV   Kbps    kbps  Cells Sts
                                             Accept

ABC-high       1/107   7          7          - / Yes  PV   10000    5000   32   UP
ABC-med        1/105   6          6          - / Yes  PV   10000                UP
ABC-low        1/102   5-0        5-0        - / Yes  -    10000                UP
```

Configuring Multi-VLAN to VC Support

For information on configuring multi-VLAN to VC support, see the "Configuring QoS for ATM VC Access Trunk Emulation" topic at http://www.cisco.rw/univercd/cc/td/doc/product/core/cis7600/cfgnotes/flexport/combo/flexqos.htm#wp1162305.

Configuring Link Fragmentation and Interleaving with Virtual Templates

The ATM SPA supports Link Fragmentation and Interleaving (LFI) with the distributed Compressed Real-Time Protocol (dCRTP). This allows the ATM interfaces, which are cell-based, to efficiently transport packet-based IP traffic without an excessive amount of bandwidth being used for packet headers and other overhead.

The LFI/dCRTP feature requires the use of multilink PPP (MLP), which can be implemented either by using virtual templates or dialer templates.

Link Fragmentation and Interleaving with Virtual Templates Configuration Guidelines

- The 1-Port OC-48c/STM-16 ATM SPA does not support LFI.
- A functional multilink PPP (MLP) bundle requires one virtual access interface operating as a PPP interface, and a second virtual access interface operating as a multilink PPP bundle interface.
- The Cisco IOS software supports a maximum of 1,000 virtual template interfaces per Cisco 7600 series router.
- When LFI is configured on a PVC, the output packets counter in the **show atm pvc** command counts all fragments of a packet as a single packet, and does not display the actual number of fragmented packets that were output. For example, if a packet is fragmented into four fragments, the output

packets counter shows only one packet, not four. The output bytes counter is accurate, however, and you can also display the total number of fragmented packets on all PVCs on the interface with the **show interface atm** command.

- LFI supports three protocol formats: AAL5CISCOPP, AAL5MUX, and AAL5SNAP
- For fragmentation to function, a QoS service policy having a minimum of two QoS queues needs to be applied to the virtual template interface.

Link Fragmentation and Interleaving with Virtual Templates Configuration Task

To configure LFI with virtual templates, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface virtual-template** *number*	Creates a virtual template and enters interface configuration mode. • *number*—Arbitrary value to identify this virtual template.
Step 2	Router(config-if)# **bandwidth** *value*	Specifies the bandwidth, in kbps, for the interfaces that use this virtual template: • *value*—Bandwidth, in kilobits per second, for the interface.
Step 3	Router(config-if)# **service-policy input** *policy-name*	Attaches the specified policy map to the input interface that uses this virtual template: • *policy-name*—Name of the policy map that was created by the **policy-map** command to be used.
Step 4	Router(config-if)# **service-policy output** *policy-name*	Attaches the specified policy map to the output interface that uses this virtual template: • *policy-name*—Name of the policy map that was created by the **policy-map** command to be used.
Step 5	Router(config-if)# **ppp multilink** [**bap**]	Enables multilink PPP (MLP) on the interfaces that use this virtual template: • **bap**—(Optional) Enables bandwidth allocation control negotiation and dynamic allocation of bandwidth on a link, using the bandwidth allocation protocol (BAP).
Step 6	Router(config-if)# **ppp multilink fragment delay** *max-delay*	(Optional) Configures the maximum delay for the transmission of a packet fragment on an MLP bundle. • *max-delay*—Maximum amount of time, in milliseconds, that should be required to transmit a fragment. The range is from 1 to 1000, with a default value of 30 for MLP bundles.
Step 7	Router(config-if)# **ppp multilink interleave**	Enables interleaving of the fragments of larger packets on an MLP bundle.
Step 8	Router(config-if)# **interface atm** *slot/subslot/port.subinterface* **point-to-point**	Creates the specified point-to-point subinterface and enters interface configuration mode.

	Command or Action	Purpose
Step 9	Router(config-if)# **pvc** [*name*] *vpi*/*vci* [**ilmi** | **qsaal**]	Configures a new ATM PVC by assigning its VPI/VCI numbers and enters ATM VC configuration mode. The valid values for *vpi*/*vci* are: • *vpi*—Specifies the VPI ID. The valid range is 0 to 255. • *vci*—Specifies the VCI ID. The valid range is 1 to 65535. Values 1 to 31 are reserved and should not be used, except for 5 for the QSAAL PVC and 16 for the ILMI PVC. You can also configure the following options: • *name*—(Optional) An arbitrary string that identifies this PVC. • **ilmi**—(Optional) Configures the PVC to use ILMI encapsulation (default). • **qsaal**—(Optional) Configures the PVC to use QSAAL encapsulation.
	Note When using the **pvc** command, remember that the *vpi*/*vci* combination forms a unique identifier for the interface and all of its subinterfaces. If you specify a *vpi*/*vci* combination that has been used on another subinterface, the Cisco IOS software assumes that you want to modify that PVC's configuration and automatically switches to its parent subinterface.	
Step 10	Router(config-if-atm-vc)# **protocol ppp virtual-template** *number*	Configures the PVC for PPP with the parameters from the specified virtual template.
Step 11	Router(config-if-atm-vc)# **end**	Exits ATM VC configuration mode and returns to privileged EXEC mode.

Verifying the Link Fragmentation and Interleaving with Virtual Templates Configuration

To verify a virtual template configuration, display the running configuration for the configured ATM and virtual interfaces:

```
Router# show running-config interface virtual-template 1

!
interface Virtual-Template1
Current configuration : 373 bytes
!
interface Virtual-Template1
bandwidth 300
ip address 23.0.0.1 255.255.255.0
ppp chap hostname template1
ppp multilink
ppp multilink fragment-delay 8
ppp multilink interleave
service-policy output lfiqos
!

Router# show running-config interface atm 6/0/1

!
interface ATM6/0/1
 atm idle-cell-format itu
 atm enable-payload-scrambling
```

```
 no atm ilmi-keepalive
 pvc 32/32
  vbr-rt 640 640 256
  encapsulation aal5snap
  protocol ppp Virtual-Template1
```

To display run-time statistics and other information about the currently configured multilink PPP bundles, use the **show ppp multilink** command:

```
Router# show ppp multilink

Virtual-Access3, bundle name is north-2
  Bundle up for 00:01:51
  Bundle is Distributed
  0 lost fragments, 0 reordered, 0 unassigned
  0 discarded, 0 lost received, 1/255 load
  0x0 received sequence, 0x0 sent sequence
  Member links: 1 (max not set, min not set)
    Vi1, since 00:01:38, no frags rcvd, 62 weight, 54 frag size

dLFI statistics:
            DLFI Packets    Pkts In             Pkts Out
              Fragmented 4294967288             3129990
            UnFragmented    1249071                   0
             Reassembled    1249071             1564994

        Reassembly Drops          0
     Fragmentation Drops          0
        Out of Seq Frags          0
```

Note The **show ppp multilink** command displays only the packet counters, and not byte counters, for a dLFI configuration on an ATM SPA interface. Also, the number of fragmented packets shows the number of fragments sent to the SAR assembly, not the number of fragments that are placed on the ATM line. It is possible that the SAR assembly might drop some of these fragments on the basis of Layer 3 QoS limits.

Configuring the Distributed Compressed Real-Time Protocol

The distributed Compressed Real-Time Protocol (dCRTP) compresses the 40 bytes of the IP/UDP/RTP packet headers down to between only two and four bytes in a distributed fast-switching and distributed Cisco Express Forwarding (dCEF) network. This compression reduces the packet size, improves the speed of packet transmission, and reduces packet latency, especially on cell-based interfaces, such as ATM interfaces.

Distributed Compressed Real-Time Protocol Configuration Guidelines

When configuring dCRTP, consider the following guidelines:

- Distributed CEF switching or distributed fast switching must be enabled on the interface.
- PPP must be used on the interface or subinterface.

Distributed Compressed Real-Time Protocol Configuration Task

To enable and configure dCRTP on an ATM interface, virtual template interface, or a dialer template interface, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port* or Router(config)# **interface virtual-template** *number* or Router(config)# **interface dialer** *number*	Enters interface configuration mode for an interface on the ATM SPA, or for a virtual template or dialer template interface.
Step 2	Router(config-if)# **ip rcp header-compression** [**passive**]	Enables RCP header compression. • **passive**—(Optional) Compresses outgoing RCP packets only if incoming RCP packets on the same interface are compressed. The default compresses all RCP packets on the interface.
Step 3	Router(config-if)# **ip tcp header-compression** [**passive**]	Enables TCP header compression. • **passive**—(Optional) Compresses outgoing TCP packets only if incoming TCP packets on the same interface are compressed. The default compresses all TCP packets on the interface.
	Note By default, RCP and TCP header compression are enabled on ATM interfaces when they are configured with an IP address. You do not need to give the **ip rcp header-compression** and **ip tcp header-compression** commands unless you have previously disabled these features, or you want to use the **passive** options.	
Step 4	Router(config-if)# **ip rcp compression-connections** *number*	Specifies the total number of RCP header compression connections that can be supported on the interface. • *number*—Number of RCP header compression connections. The valid range is 3 to 1000, with a default of 32 connections (16 calls).
Step 5	Router(config-if)# **ip tcp compression-connections** *number*	Specifies the total number of TCP header compression connections that can be supported on the interface. • *number*—Number of TCP header compression connections. The valid range is 3 to 1000, with a default of 32 connections (16 calls).
Step 6	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying the Distributed Compressed Real-Time Protocol Configuration

To verify the dCRTP of an ATM interface, use the **show running-config interface interface virtual-template** command:

```
Router# show running-config interface interface virtual-template 1

!
interface Virtual-Template1
 bandwidth 2320
 ip unnumbered Loopback2
 max-reserved-bandwidth 100
```

```
ip tcp header-compression
ppp multilink
ppp multilink fragment delay 4
ppp multilink interleave
ip rtp header-compression
```

Configuring Automatic Protection Switching

The ATM SPAs support 1+1 Automatic Protection Switching (APS) on PVCs as described in section 5.3 of the Telcordia publication *GR-253-CORE SONET Transport Systems: Common Generic Criteria*. APS redundancy is supported at the line layer, so that when an OC-3c, OC-12c, or OC-48c link fails, all of the PVCs that are carried by that link are switched simultaneously.

Note APS is not supported for SVCs.

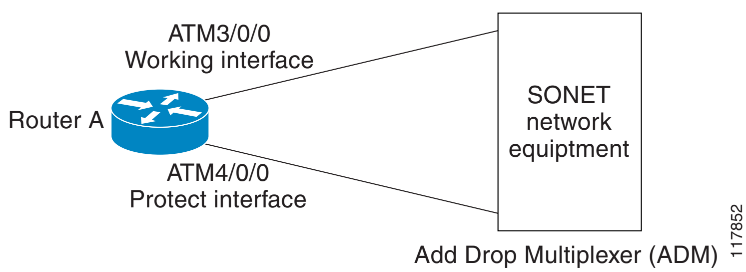
In an APS configuration, a redundant ATM interface (the Protect interface) is configured for every active ATM interface (the Working interface). If the Working interface goes down, the Protect interface automatically switches over and continues communication over the interface's PVCs.

The APS Protect Group Protocol (PGP), which runs on top of User Datagram Protocol (UDP), provides communication between the Working and Protect interfaces. This communication occurs over a separate out-of-band (OOB) communication channel, such as an Ethernet link.

In the case of degradation, loss of channel signal, or manual intervention, the APS software on the Protect interface sends APS PGP commands to activate or deactivate the Working interface as necessary. If the communication channel between the Working and Protect interfaces is lost, the Working interface assumes full control, as if no Protect interface existed.

Figure 7-4 shows a very simple example of a pair of Working and Protect interfaces on a single router.

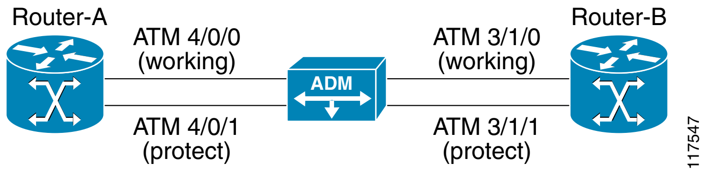
Figure 7-4 Basic Automatic Protection Switching Configuration



Tip If possible, use separate SPAs to provide the Working and Protect interfaces, as shown in Figure 7-4. This removes the SPA as a potential single point of failure, which would be the case if the same SPA provided both the Working and Protect interfaces.

Multiple routers can be using APS at the same time. For example, Figure 7-5 shows a simple example of two routers that each have one pair of Working and Protect interfaces. In this configuration, the two routers are independently configured.

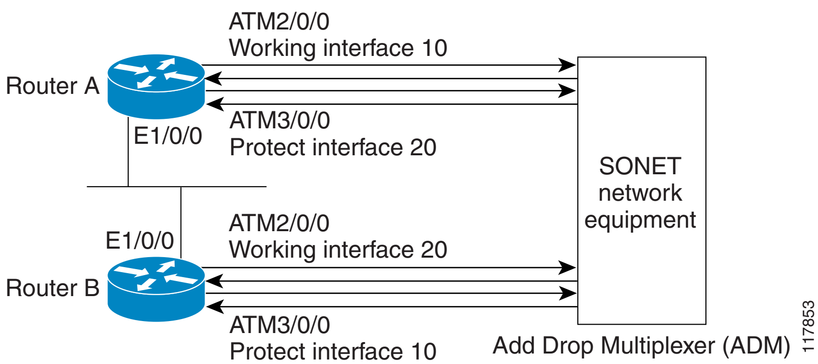
Figure 7-5 Sample Automatic Protection Switching Configuration with Multiple Routers



You can also configure multiple routers with APS so that interfaces on one router can provide protection for the interfaces on another router. This provides protection in case a router experiences a major system problem, such as a processor fault.

Figure 7-6 shows a basic example of two routers that each have one Working ATM interface. Each router also has one Protect interface that provides protection for the other router's Working interface. Note that this configuration requires a separate out-of-band (OOB) communication link between the two routers, which in this case is provided by the Ethernet network.

Figure 7-6 Sample Multiple Router Protection with Automatic Protection Switching



An APS configuration requires the following steps:

- Configure the Working interface with the desired IP addresses, subinterfaces, and PVCs. Also assign the interface to an APS group and designate it as the Working interface.
- Create a loopback circuit for communication between the Working and Protect interfaces. This is optional, because you can also use any valid IP address on the router. However, we recommend using a loopback interface because it is always up and provides connectivity between the two interfaces as long as any communication path exists between them.
- Configure the Protect interface with the same subinterfaces and PVCs that were configured on the Working interface. The Protect interface should also be configured with an IP address that is on the same subnet as the Working interface.

Tip Always configure the Working interface before the Protect interface, so as to prevent the Protect interface from becoming active and disabling the circuits on the Working interface.

Automatic Protection Switching Configuration Guidelines

When configuring APS, consider the following guidelines:

- The Working and Protect interfaces must be compatible (that is, both OC-3c or both OC-12c interfaces). The interfaces can be on the same SPA, different SPAs in the same router, or different SPAs in different routers.
- If using interfaces on different routers, the two routers must have a network connection other than the ATM connection (such as through an Ethernet LAN). Because the APS PGP is UDP traffic, this network connection should be reliable with a minimum number of hops.
- Configure the Working ATM interface with the desired IP addresses and other parameters, as described in the "Required Configuration Tasks" section on page 7-2 and the "Configuring SONET and SDH Framing" section on page 7-57.
- Configure the desired PVCs on the Working interface, as described in the different procedures that are listed in the "Creating a Permanent Virtual Circuit" section on page 7-7.
- The IP addresses on the Working and Protect interfaces should be in the same subnet.
- APS is not supported on SVCs.

Automatic Protection Switching Configuration Task

To configure the Working and Protect interfaces on the ATM SPAs for basic APS operation, perform the following procedure beginning in global configuration mode. For complete information on APS, including information on additional APS features, refer to the "Configuring ATM Interfaces" chapter in the *Cisco IOS Interface Configuration Guide, Release 12.2*.

	Command or Action	Purpose
Step 1	Router(config)# **interface loopback** *interface-number*	Creates a loopback interface and enters interface configuration mode: • *interface-number*—An arbitrary value from 0 to 2,147,483,647 that uniquely identifies this loopback interface.
Step 2	Router(config-if)# **ip address** *ip-address mask* [**secondary**]	Specifies the IP address and subnet mask for this loopback interface. If the Working and Protect interfaces are on the same router, this IP address should be in the same subnet as the Working interface. If the Working and Protect interfaces are on different routers, this IP address should be in the same subnet as the Ethernet interface that provides the connectivity between the two routers. Repeat this command with the **secondary** keyword to specify additional IP addresses to be used for this interface.
Step 3	Router(config-if)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the Working interface on the ATM SPA.
Step 4	Router(config-if)# **ip address** *ip-address mask* [**secondary**]	Specifies the IP address and subnet mask for the Working interface. Repeat this command with the **secondary** keyword to specify additional IP addresses to be used for the interface.

	Command or Action	Purpose
Step 5	Router(config-if)# **aps group** *group-number*	Enables the use of the APS Protect Group Protocol for this Working interface. • *group-number*—Unique number identifying this pair of Working and Protect interfaces. **Note** The **aps group** command is optional if this is the only pair of Working and Protect interfaces on the router, but is required when you configure more than one pair of Working and Protect interfaces on the same router.
Step 6	Router(config-if)# **aps working** *circuit-number*	Identifies the interface as the Working interface. • *circuit-number*—Identification number for this particular channel in the APS pair. Because only 1+1 redundancy is supported, the only valid values are 0 or 1, and the Working interface defaults to 1.
Step 7	Router(config-if)# **aps authentication** *security-string*	(Optional) Specifies a security string that must be included in every OOB message sent between the Working and Protect interfaces. • *security-string*—Arbitrary string to be used as a password between the Working and Protect interfaces. This string must match the one configured on the Protect interface.
Step 8	Router(config-if)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the Protect interface on the ATM SPA.
Step 9	Router(config-if)# **ip address** *ip-address mask* [**secondary**]	Specifies the IP address and subnet mask for the Protect interface. **Note** This should be the same address that was configured on the Working interface in Step 4. Repeat this command with the **secondary** keyword to specify additional IP addresses to be used for the interface. These should match the secondary IP addresses that are configured on the Working interface.
Step 10	Router(config-if)# **aps group** *group-number*	Enables the use of the APS Protect Group Protocol for this Protect interface. • *group-number*—Unique number identifying this pair of Working and Protect interfaces. **Note** The **aps group** command is optional if this is the only pair of Working and Protect interfaces on the router, but is required when you configure more than one pair of Working and Protect interfaces on the same router.

	Command or Action	Purpose
Step 11	Router(config-if)# **aps protect** *circuit-number ip-address*	Identifies this interface as the Protect interface: • *circuit-number*—Identification number for this particular channel in the APS pair. Because only 1+1 redundancy is supported, the only valid values are 0 or 1, and the Protect interface defaults to 0. • *ip-address*—IP address for the loopback interface that was configured in Step 2. The Protect interface uses this IP address to communicate with the Working interface. **Note** If you do not want to use a loopback interface for this configuration, this IP address should be the address of the Working interface if the Protect and Working interfaces are on the same router. If the Working and Protect interfaces are on different routers, this should be the IP address of the Ethernet interface that provides interconnectivity between the two routers.
Step 12	Router(config-if)# **aps authentication** *security-string*	(Optional) Specifies a security string that must be included in every OOB message sent between the Working and Protect interfaces. • *security-string*—Arbitrary string to be used as a password between the Working and Protect interfaces. This string must match the one configured on the Working interface.
Step 13	Router(config-if)# **aps revert** *minutes*	(Optional) Enables the Protect interface to automatically switch back to the Working interface after the Working interface has been up for a specified number of minutes. • *minutes*—Number of minutes until the interface is switched back to the Working interface after the Working interface comes back up. **Note** If this command is not given, you must manually switch back to the Working interface using either the **aps force** *circuit-number* or the **aps manual** *circuit-number* command.
Step 14	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying the Automatic Protection Switching Configuration

To verify the APS configuration on the router, use the **show aps** command without any options. The following example shows a typical configuration in which the Working interface is the active interface:

```
Router# show aps

ATM4/0/1 APS Group 1: protect channel 0 (inactive)
        bidirectional, revertive (2 min)
        PGP timers (default): hello time=1; hold time=3
        state:
        authentication = (default)
```

```
        PGP versions (native/negotiated): 2/2
        SONET framing; SONET APS signalling by default
        Received K1K2: 0x00 0x05
                No Request (Null)
        Transmitted K1K2: 0x20 0x05
                Reverse Request (protect)
        Working channel 1 at 10.10.10.41 Enabled
        Remote APS configuration: (null)

ATM4/0/0 APS Group 1: working channel 1 (active)
        PGP timers (from protect): hello time=3; hold time=6
        state: Enabled
        authentication = (default)
        PGP versions (native/negotiated): 2/2
        SONET framing; SONET APS signalling by default
        Protect at 10.10.10.41
        Remote APS configuration: (null)
```

The following sample output is for the same interfaces, except that the Working interface has gone down and the Protect interface is now active:

```
Router# show aps

ATM4/0/1 APS Group 1: protect channel 0 (active)
        bidirectional, revertive (2 min)
        PGP timers (default): hello time=1; hold time=3
        state:
        authentication = (default)
        PGP versions (native/negotiated): 2/2
        SONET framing; SONET APS signalling by default
        Received K1K2: 0x00 0x05
                No Request (Null)
        Transmitted K1K2: 0xC1 0x05
                Signal Failure - Low Priority (working)
        Working channel 1 at 10.10.10.41 Disabled SF
        Pending local request(s):
                0xC (, channel(s) 1)
        Remote APS configuration: (null)

ATM4/0/0 APS Group 1: working channel 1 (Interface down)
        PGP timers (from protect): hello time=3; hold time=6
        state: Disabled
        authentication = (default)
        PGP versions (native/negotiated): 2/2
        SONET framing; SONET APS signalling by default
        Protect at 10.10.10.41
        Remote APS configuration: (null)
```

Tip To obtain APS information for a specific ATM interface, use the **show aps atm** *slot/subslot/port* command. To display information about the APS groups that are configured on the router, use the **show aps group** command.

Configuring SONET and SDH Framing

The default framing on the ATM OC-3c and OC-12c SPAs is SONET, but the interfaces also support SDH framing.

Note In ATM environments, the key difference between SONET and SDH framing modes is the type of cell transmitted when no user or data cells are available. The ATM forum specifies the use of idle cells when unassigned cells are not being generated. More specifically, in Synchronous Transport Module-X (STM-X) mode, an ATM interface sends idle cells for cell-rate decoupling. In Synchronous Transport Signal-Xc (STS-Xc) mode, the ATM interface sends unassigned cells for cell-rate decoupling.

To change the framing type and configure optional parameters, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the indicated port on the specified ATM SPAs.
Step 2	Router(config-if)# **atm clock internal**	(Optional) Configures the interface to use its own internal (onboard) clock to clock transmitted data. The default (**no atm clock internal**) configures the interface to use the transmit clock signal that is recovered from the receive data stream, allowing the switch to provide the clocking source.
Step 3	Router(config-if)# **atm framing** {**sdh** | **sonet**}	(Optional) Configures the interface for either SDH or SONET framing. The default is SONET.
Step 4	Router(config-if)# [**no**] **atm sonet report** {**all** | **b1-tca** | **b2-tca** | **b3-tca** | **default** | **lais** | **lrdi** | **pais** | **plop** | **pplm** | **prdi** | **ptim** | **puneq** | **sd-ber** | **sf-ber** | **slof** | **slos**}	(Optional) Enables ATM SONET alarm reporting on the interface. The default is for all reports to be disabled. You can enable an individual alarm, or you can enable all alarms with the **all** keyword. **Note** This command also supports a **none** [**ignore**] option, which cannot be used with any of the other options. See the "Configuring for Transmit-Only Mode" section on page 7-58 for details.
Step 5	Router(config-if)# [**no**] **atm sonet-threshold** {**b1-tca** *value* | **b2-tca** *value* | **b3-tca** *value* | **sd-ber** *value* | **sf-ber** *value*}	(Optional) Configures the BER threshold values on the interface. The value specifies a negative exponent to the power of 10 (10 to the power of minus *value*) for the threshold value. The default values are the following: • **b1-tca** = 6 (10e–6) • **b2-tca** = 6 (10e–6) • **b3-tca** = 6 (10e–6) • **sd-ber** = 6 (10e–6) • **sf-ber** = 3 (10e–3)
Step 6	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying the SONET and SDH Framing Configuration

To verify the framing configuration, use the **show controllers atm** command:

```
Router# show controllers atm 5/0/1

Interface ATM5/0/1 is up
 Framing mode: SONET OC3 STS-3c

SONET Subblock:
SECTION
  LOF = 0          LOS    = 0                            BIP(B1) = 603
LINE
  AIS = 0          RDI    = 2          FEBE = 2332       BIP(B2) = 1018
PATH
  AIS = 0          RDI    = 1          FEBE = 28         BIP(B3) = 228
  LOP = 0          NEWPTR = 0          PSE  = 1          NSE     = 2

Active Defects: None
Active Alarms:  None
Alarm reporting enabled for: LOF LOS B1-TCA B2-TCA SF LOP B3-TCA

ATM framing errors:
  HCS (correctable):   0
  HCS (uncorrectable): 0

APS

  COAPS = 0          PSBF = 0
  State: PSBF_state = False
  Rx(K1/K2): 00/00  Tx(K1/K2): 00/00
  Rx Synchronization Status S1 = 00
  S1S0 = 00, C2 = 00

PATH TRACE BUFFER : STABLE

BER thresholds:  SF = 10e-3  SD = 10e-6
TCA thresholds:  B1 = 10e-7 B2 = 10e-6  B3 = 10e-6

  Clock source:  line
```

Configuring for Transmit-Only Mode

The ATM SPAs support operation in a transmit-only mode, where a receive fiber does not need to be connected. This mode is typically used for one-way applications, such as video-on-demand.

By default, the lack of a receive path generates continuous framing errors, which bring the ATM interface down. To prevent this, you must configure the ATM interface to disable and ignore all ATM SONET alarms. The 1-Port OC-48c/STM-16 ATM SPA default framing is SONET.

Note This configuration violates the ATM specifications for alarm reporting.

Transmit-Only Mode Configuration Guidelines

When an ATM interface has been configured to ignore ATM SONET alarms, you cannot configure an IP address (or other Layer 3 parameter) on the interface. Similarly, you must remove all IP addresses (and all other Layer 3 parameters) from the interface before beginning this procedure.

Transmit-Only Mode Configuration Task

To configure the ATM interface to disable and ignore all ATM SONET alarms, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot*/*subslot*/*port*[.*subinterface*]	Enters interface (or subinterface) configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-if)# **no ip address** *ip-address mask*	Removes the IP address that is assigned to this interface (if one has been configured). All IP and other Layer 3 configurations must be removed from the interface before ATM SONET alarms can be ignored.
Step 3	Router(config-if)# **atm sonet report none ignore**	Disables the generation of all ATM SONET alarms, and instructs the ATM interface to remain up and operational when such alarm conditions exist.
Step 4	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring AToM VP Cell Mode Relay Support

To configure Any Transport over MPLS (AToM) Cell Mode Relay, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot*/*subslot*/*port*	Enters interface configuration mode for the indicated port on the specified ATM SPA
Step 2	Router(config-if)# **no ip address** *ip-address mask*	Removes the IP address that is assigned to this interface (if one has been configured).
Step 3	Router(config-if)# **atm pvp vpi l2transport**	Creates a permanent virtual path (PVP) used to multiplex (or bundle) one or more virtual circuits (VCs).
Step 4	Router(config-if)# **xconnect peer-router-id vcid encapsulation mpls**	Routes a Layer 2 packets over a specified point-to-point VC by using Ethernet over multiprotocol label switching (EoMPLS).
Step 5	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

VP Mode Configuration Guidelines

When configuring ATM Cell Relay over MPLS in VP mode, use the following guidelines:

- You do not need to enter the **encapsulation aal0** command in VP mode.
- One ATM interface can accommodate multiple types of ATM connections. VP cell relay, VC cell relay, and ATM AAL5 over MPLS can coexist on one ATM interface.
- If a VPI is configured for VP cell relay, you cannot configure a PVC using the same VPI.

- VP trunking (mapping multiple VPs to one emulated VC label) is not supported in this release. Each VP is mapped to one emulated VC.
- Each VP is associated with one unique emulated VC ID. The AToM emulated VC type is ATM VP Cell Transport.
- The AToM control word is supported. However, if a peer PE does not support the control word, it is disabled. This negotiation is done by LDP label binding.
- VP mode (and VC mode) drop idle cells.

VP Mode Configuration Example

The following example transports single ATM cells over a virtual path:

```
Router# pseudowire-class vp-cell-relay
encapsulation mpls
int atm 5/0
atm pvp 1 l2transport
xconnect 10.0.0.1 123 pw-class vp-cell-relay
```

Verifying ATM Cell Relay VP Mode

The following **show atm vp** command shows that the interface is configured for VP mode cell relay:

```
Router# show atm vp 1
ATM5/0  VPI: 1, Cell Relay, PeakRate: 149760, CesRate: 0, DataVCs: 1, CesVCs: 0, Status:
ACTIVE
VCD    VCI    Type   InPkts    OutPkts   AAL/Encap     Status
6      3      PVC    0         0         F4 OAM        ACTIVE
7      4      PVC    0         0         F4 OAM        ACTIVE
TotalInPkts: 0, TotalOutPkts: 0, TotalInFast: 0, TotalOutFast: 0,
TotalBroadcasts: 0 TotalInPktDrops: 0, TotalOutPktDrops: 0
```

Configuring QoS Features on ATM SPAs

The SIPs and SPAs support many QoS features using modular QoS CLI (MQC) configuration. For information about the QoS features supported by the ATM SPAs, see the "Configuring QoS Features on a SIP" section on page 4-61 of Chapter 4, "Configuring the SIPs and SSC."

ATM SPA QoS Configuration Guidelines

For the 2-Port and 4-Port OC-3c/STM-1 ATM SPA, the following applies:

- In the ingress direction, all Quality of Service (QoS) features are supported by the Cisco 7600 SIP-200.
- In the egress direction:
  - All queueing-based features (such as class-based weighted fair queueing [CBWFQ], and ATM per-VC WFQ, WRED, and shaping) are implemented on the segmentation and reassembly (SAR) processor on the SPA.
  - Policing is implemented on the SIP.
  - Class queue shaping is not supported.

Phase 2 Access Circuit Redundancy with Local Switching

Phase 2 Access Circuit Redundancy with Local Switching provides a backup attachment circuit (AC) when the primary attachment circuit fails. All the ACs must be on same Cisco 7600 series router.

The following combinations of ATM ACs are supported:

- ATM ACs on the same SPA
- ATM ACs on different SPAs on the same SIP
- ATM ACs on different SIPs on the same Cisco 7600 series router

Note For Cisco IOS release 12.2(33)SRC, this feature is supported on the 24-Port Channelized T1/E1 ATM CEoP SPA and the 1-Port Channelized OC-3 STM1 ATM CEoP SPA, as well as the 2-Port and 4-Port OC-3c/STM-1 ATM SPA, the 1-Port OC-12c/STM-4 ATM SPA, and the 1-Port OC-48c/STM-16 ATM SPA.

Guidelines

- Autoconfiguration of ATM interfaces is supported.
- Only the tail end AC can be backed up, if head end fails there is no protection.
- The circuit type of the primary and backup AC must be identical (failover operation will not switch between different types of interfaces or different CEM circuit types).
- Only one backup AC is allowed for each connection.
- Autoconfiguration is allowed for backup ATM Permanent Virtual Circuits (PVCs) or ATM Permanent Virtual Paths (PVPs) .
- The ATM circuit used as a backup in a local switching connection cannot be used for xconnect configurations.
- Dynamic modification of parameters in a local switching connection is not supported in the case where the tail-end segment is backed up to a segment using the **backup** command. If you want to modify the parameters in any of the three segments (head-end, tail-end, or backup segment), you must first unconfigure with the **backup** command, make the changes in the individual segments, and then re-configure the backup with the **backup** command.

Configuration

	Command or Action	Purpose
Step 1	`Router(config)# [no] connect name atma/b/c vpi/vci atmx/y/z vpi/vci`	Configures a local switching connection between two ATM interfaces. The **no** form of this command unconfigures a local switching connection between two ATM interfaces.
	`Router(config-connection)# backup interface atm x/y/z vpi/vci`	Backs up a locally switched ATM connection.

Configuration Example

```
Router(config)# connect ATM atm2/0/0 0 atm3/0/0 0
Router(config-connection)# backup interface atm4/0/0 1
```

Verifying

Use the **show xconnect all** command to check the status of the backup and primary circuits.

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), use the following command in privileged EXEC configuration mode:

Note To permanently save your configuration changes, you must write them to the nonvolatile RAM (NVRAM) by entering the **copy running-config startup-config** command in privileged EXEC mode.

Command	Purpose
Router# **copy running-config startup-config**	Writes the new configuration to NVRAM.

For more information about managing configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

Shutting Down and Restarting an Interface on a SPA

Shutting down an interface puts it into the administratively down mode and takes it offline, stopping all traffic that is passing through the interface. Shutting down an interface, though, does not change the interface configuration.

As a general rule, you do not need to shut down an interface if you are removing it and replacing it with the same exact model of SPA in an online insertion and removal (OIR) operation. However, we recommend shutting down an interface whenever you are performing one of the following tasks:

- When you do not need to use the interface in the network.
- Preparing for future testing or troubleshooting.
- Changing the interface configuration in a way that would affect the traffic flow, such as changing the encapsulation.
- Changing the interface cables.
- Removing a SPA that you do not expect to replace.
- Replacing the SIP with another type of SIP (such as replacing a Cisco 7600 SIP-200 with a Cisco 7600 SIP-400).
- Replacing an interface card with a different model of card.

Shutting down the interface in these situations prevents anomalies from occurring when you reinstall the new card or cables. It also reduces the number of error messages and system messages that might otherwise appear.

Tip If you are planning on physically removing the SPA from the SIP, also shut down the SPA, using the procedure given in the "Shutting Down an ATM Shared Port Adapter" section on page 7-64.

Note If you plan to replace an existing ATM port adapter with an ATM SPA in the Cisco 7600 series router and want to use the same configuration, save the slot's configuration before physically replacing the hardware. This is because all slot configuration is lost when you replace one card type with another card type, even if the two cards are functionally equivalent. You can then re-enter the previous configuration after you have inserted the ATM SPA.

To shut down an interface, perform the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the indicated port on the specified ATM SPA.
Step 2	Router(config-if)# **shutdown**	Shuts down the interface.
	Note Repeat Step 1 and Step 2 for each interface to be shut down.	
Step 3	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Tip When you shut down an interface, the **show interface** command indicates that the interface is administratively down until the SPA is physically removed from the chassis or until the SPA is re-enabled.

The following shows a typical example of shutting down an ATM SPA interface:

```
Router> enable
Router# configure terminal
Router(config)# interface atm 4/0/0
Router(config-if)# shutdown
Router(config-if)# end
Router# show interface atm 4/0/0

ATM4/0/0 is administratively down, line protocol is down
  Hardware is SPA-4XOC3-ATM, address is 000d.2959.d5ca (bia 000d.2959.d5ca)
  Internet address is 10.10.10.16/24
  MTU 4470 bytes, sub MTU 4470, BW 599040 Kbit, DLY 80 usec,
     reliability 255/255, txload 42/255, rxload 1/255
  Encapsulation ATM, loopback not set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 1 current VCCs
  VC idle disconnect time: 300 seconds
  0 carrier transitions
  Last input 01:01:16, output 01:01:16, output hang never
  Last clearing of "show interface" counters 01:10:21
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
```

```
30 second input rate 0 bits/sec, 0 packets/sec
30 second output rate 702176000 bits/sec, 1415679 packets/sec
   1000 packets input, 112000 bytes, 0 no buffer
   Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
   0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
   2948203354 packets output, 182788653886 bytes, 0 underruns
   0 output errors, 0 collisions, 0 interface resets
   0 output buffer failures, 0 output buffers swapped out
```

Shutting Down an ATM Shared Port Adapter

Shutting down an ATM SPA shuts down all ATM interfaces on the SPA, and puts the SPA and its interfaces into the administratively down state. This takes all interfaces offline, stopping all traffic that is passing through the SPA. Shutting down an ATM SPA, though, does not change the configuration of the SPA and its interfaces.

As a general rule, you do not need to shut down an ATM SPA if you are removing it and replacing it with the same exact model of SPA in an online insertion and removal (OIR) operation. However, you should shut down the ATM SPA whenever you are performing one of the following tasks:

- Removing an interface that you do not expect to replace.
- Replacing the SIP with another type of SIP (such as replacing a Cisco 7600 SIP-200 with a Cisco 7600 SIP-400).
- Replacing the ATM SPA with a different model of SPA.

To shut down the ATM SPA, use the following procedure beginning in global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **hw-module subslot** *slot/subslot* **shutdown** [**powered** | **unpowered**]	Shuts down the ATM SPA. • **powered**—(Optional) Shuts down the ATM SPA and leaves it in the reset state. This is the default and is typically done when you want to shut down the SPA but leave it physically installed and cabled in the Cisco 7600 series router. • **unpowered**—(Optional) Shuts down the ATM SPA and leaves it in the unpowered state. Typically, this is done before removing the ATM SPA from the chassis. **Note** Repeat this step for each ATM SPA to be shut down.
	Note The **hw-module subslot shutdown** command can be given in both the global configuration and privileged EXEC modes. If this command is given in global configuration mode, it can be saved to the startup configuration so that it is automatically executed after each reload of the router. If given in privileged EXEC mode, the command takes effect immediately, but it is not saved to the configuration. In either case, the **hw-module subslot shutdown** command remains in effect during the current session of the Cisco 7600 series router until it is reversed using the **no** form of the command.	
Step 2	Router(config)# **end**	Exits configuration mode and returns to privileged EXEC mode.

The following shows a typical example of shutting down ATM SPAs. In this example, the SPA in subslot 0 is put into reset mode, while the SPA in subslot 1 is powered down.

```
Router> enable
Router# hw-module subslot 4/0 shutdown powered
Router# hw-module subslot 4/1 shutdown unpowered
```

Tip The ATM SPA remains shut down, even after a new SPA is installed or after a reset of the Cisco 7600 series router, until you re-enable the SPA using the **no hw-module subslot shutdown** command.

Verifying the Interface Configuration

See the following sections to obtain configuration and operational information about the ATM SPA and its interfaces:

- Verifying Per-Port Interface Status, page 7-65
- Monitoring Per-Port Interface Statistics, page 7-66

For additional information on using these and other commands to obtain information about the configuration and operation of the ATM SPAs and interfaces, see Chapter 8, "Troubleshooting the ATM Shared Port Adapter."

Verifying Per-Port Interface Status

Use the **show interfaces atm** command to display detailed status information about an interface port in an ATM SPA that is installed in the Cisco 7600 series router. The following example provides sample output for interface port 1 (the second port) on the ATM SPA that is located in subslot 0 (the left-most subslot), of the SIP that is installed in slot 3 of a Cisco 7600 series router:

```
Router# show interface atm 3/0/1

ATM3/0/1 is up, line protocol is up
  Hardware is SPA-4XOC3-ATM, address is 000a.f330.7dc0 (bia 000a.f330.7dca)
  Internet address is 10.13.21.31/24
  MTU 4470 bytes, sub MTU 4470, BW 599040 Kbit, DLY 80 usec,
     reliability 255/255, txload 140/255, rxload 129/255
  Encapsulation ATM, loopback not set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 1 current VCCs
  VC idle disconnect time: 300 seconds
  0 carrier transitions
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 00:45:35
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 304387000 bits/sec, 396342 packets/sec
  5 minute output rate 329747000 bits/sec, 396334 packets/sec
     1239456438 packets input, 118987818048 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     1239456287 packets output, 128903453848 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Monitoring Per-Port Interface Statistics

Use the **show controllers atm** command to display detailed status and statistical information on a per-port basis for an ATM SPA. The following example provides sample output for interface port 0 (the first port) on the ATM SPA that is located in subslot 0 (the left-most subslot) of the SIP that is installed in slot 4 of a Cisco 7600 series router:

```
Router# show controllers atm 4/0/0

Interface ATM4/0/0 is up
 Framing mode: SONET OC3 STS-3c

SONET Subblock:
SECTION
  LOF = 0          LOS    = 0                            BIP(B1) = 603
LINE
  AIS = 0          RDI    = 2          FEBE = 2332       BIP(B2) = 1018
PATH
  AIS = 0          RDI    = 1          FEBE = 28         BIP(B3) = 228
  LOP = 0          NEWPTR = 0          PSE  = 1          NSE     = 2

Active Defects: None
Active Alarms:  None
Alarm reporting enabled for: SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA

ATM framing errors:
  HCS (correctable):   0
  HCS (uncorrectable): 0

APS

  COAPS = 0          PSBF = 0
  State: PSBF_state = False
  Rx(K1/K2): 00/00  Tx(K1/K2): 00/00
  Rx Synchronization Status S1 = 00
  S1S0 = 00, C2 = 00

PATH TRACE BUFFER : STABLE
  Remote hostname : fecao7609_2
  Remote interface: ATM9/0/0
  Remote IP addr  : 0.0.0.0
  Remote Rx(K1/K2): 00/00  Tx(K1/K2): 00/00


BER thresholds:  SF = 10e-3  SD = 10e-6
TCA thresholds:  B1 = 10e-6  B2 = 10e-6  B3 = 10e-6

  Clock source:  line
```

Configuration Examples

This section includes the following configuration examples for the ATM SPAs:

- Basic Interface Configuration Example, page 7-67
- MTU Configuration Example, page 7-68
- Permanent Virtual Circuit Configuration Example, page 7-68
- PVC on a Point-to-Point Subinterface Configuration Example, page 7-68

Basic Interface Configuration Example

```
!
interface ATM5/1/0
 mtu 9216
 no ip address
 atm clock INTERNAL
!
interface ATM5/1/0.1 point-to-point
 mtu 9216
 ip address 70.1.1.1 255.255.0.0
 pvc 52/100
!
!
interface ATM5/1/1
 mtu 9216
 no ip address
 atm clock INTERNAL
!
interface ATM5/1/1.1 point-to-point
 mtu 9216
 ip address 70.2.1.1 255.255.0.0
 pvc 53/100
```

```
!
!
interface ATM5/1/2
 no ip address
 atm clock INTERNAL
!
interface ATM5/1/3
 no ip address
 atm clock INTERNAL
!
```

MTU Configuration Example

```
!
interface ATM4/1/0
 ip address 192.168.100.13 255.255.255.0
 mtu 9216
 ip mtu 9188
 mpls mtu 9288
 atm clock INTERNAL
!
```

Permanent Virtual Circuit Configuration Example

```
!
interface ATM5/0/0
 no ip address
 pvc 1/100
  protocol ip 1.1.1.3
  protocol ip 20.1.1.1
  broadcast
!
!
interface ATM5/0/1
 no ip address
!
interface ATM5/1/1
 ip address 1.1.1.1 255.255.255.0
 load-interval 30
 pvc 1/100
  protocol ip 1.1.1.3
  protocol ip 20.1.1.1
  cbr 140000
  broadcast
  oam-pvc manage
!
 pvc 1/101
  protocol ip 9.9.9.2
  encapsulation aal5ciscoppp Virtual-Template1
 !
```

PVC on a Point-to-Point Subinterface Configuration Example

The following example shows a simple configuration of several PVCs that are configured on point-to-point subinterfaces:

```
interface ATM3/1/0
 no ip address
!
interface ATM3/1/0.1 point-to-point
 pvc 4/44 l2transport
  mpls l2transport route 22.22.22.22 400
!
!
interface ATM3/1/0.2 point-to-point
 pvc 5/55 l2transport
  encapsulation aal0
  mpls l2transport route 22.22.22.22 500
 !
!
interface ATM3/1/0.3 point-to-point
 ip address 99.0.0.2 255.0.0.0
 pvc 9/99
 !
!
interface ATM5/0/0
 description flexwan_6_0_0
 no ip address
 logging event link-status
 atm clock INTERNAL
!
interface ATM5/0/0.1 point-to-point
 ip address 50.1.1.1 255.255.255.0
 pvc 50/11
!
!
interface ATM5/0/0.2 point-to-point
 ip address 50.2.2.1 255.255.255.0
 pvc 50/12
!
!
interface ATM5/0/0.3 point-to-point
 ip address 50.3.3.1 255.255.255.0
 pvc 50/13
 !
!
interface ATM5/0/0.4 point-to-point
 ip address 50.4.4.1 255.255.255.0
 pvc 50/14
 !
!
interface ATM5/0/0.5 point-to-point
 ip address 50.5.5.1 255.255.255.0
 pvc 50/15
 !
!
interface ATM5/1/0.1 point-to-point
 ip address 2.0.0.2 255.255.255.0
!
interface ATM5/1/0.2 point-to-point
 ip address 2.0.1.2 255.255.255.0
!
interface ATM5/1/0.3 point-to-point
 ip address 39.0.0.1 255.0.0.0
!
```

PVC on a Multipoint Subinterface Configuration Example

```
!
interface ATM4/1/0
 no ip address
 atm clock INTERNAL
!
interface ATM4/1/0.2 multipoint
 ip address 1.1.1.1 255.0.0.0
 pvc 0/121
  protocol ip 1.1.1.23 broadcast
  vbr-nrt 2358 2358
  encapsulation aal5snap
!
 pvc 0/122
  protocol ip 1.1.1.24 broadcast
  vbr-nrt 2358 2358
  encapsulation aal5snap
 !
 pvc 0/123
  protocol ip 1.1.1.25 broadcast
  vbr-nrt 2358 2358
  encapsulation aal5snap
!
 pvc 0/124
  protocol ip 1.1.1.26 broadcast
  vbr-nrt 2358 2358
  encapsulation aal5snap
!
 pvc 0/125
  protocol ip 1.1.1.27 broadcast
!
...
interface ATM5/1/1
 ip address 1.1.1.1 255.255.255.0
 load-interval 30
 pvc 1/100
  protocol ip 1.1.1.3
  protocol ip 20.1.1.1
  cbr 140000
  broadcast
  oam-pvc manage
!
 pvc 1/101
  protocol ip 9.9.9.2
  encapsulation aal5ciscoppp Virtual-Template1
!
!
interface ATM5/1/1.200 multipoint
 ip address 7.7.7.1 255.255.255.0
 bundle bundle
  pvc-bundle high 2/100
   class-vc high
  pvc-bundle med 2/101
   class-vc med
  pvc-bundle low 2/102
   class-vc low
!
!
interface ATM5/1/2
 no ip address
!
interface ATM5/1/3
```

```
 no ip address
!
```

RFC 1483 Bridging for PVCs Configuration Example

The following shows a simple example of an ATM interface and PVC that have been configured for RFC 1483 bridging with a Fast Ethernet interface:

```
vlan 30
!
interface FastEthernet7/1
 no ip address
 duplex full
 speed 100
 switchport
 switchport access vlan 30
 switchport mode access
!
interface ATM9/1/0
 no ip address
 mtu 4096
  bandwidth 2000
  pvc 0/39
 bridge-domain 30
 encapsulation aal5snap
!
interface ATM9/1/0.2 point-to-point
 ip address 10.10.12.2 255.255.255.0
 ip access-group rbe-list in
 atm route-bridged ip
 no mls ip
 pvc 10/200
!
router rip
 network 10.0.0.0
 network 30.0.0.0
!
```

RFC 1483 Bridging for PVCs with IEEE 802.1Q Tunneling Configuration Example

The following shows a simple example of an ATM interface that has been configured for RFC 1483 bridging using IEEE 802.1Q tunneling:

```
interface ATM6/2/0
 no ip address
 shutdown
 atm clock INTERNAL
 atm mtu-reject-call
 no atm ilmi-keepalive
 pvc 2/101
  bridge-domain 99 dot1q-tunnel
!
 mls qos trust dscp
 spanning-tree bpdufilter enable
```

ATM RFC 1483 Half-Bridging Configuration Example

The following simple example shows an ATM subinterface configured for half-bridging:

```
!
interface ATM5/1/0.100 multipoint
  ip address 192.168.100.14 255.255.0.0
  mtu 1500
  pvc 10/200
   encapsulation aal5snap bridge
!
```

ATM Routed Bridge Encapsulation Configuration Example

The following simple example shows an ATM subinterface configured for RBE, also known as RFC 1483 half-bridging:

```
!
interface ATM5/1/0.100 point-to-point
  ip address 10.10.10.121 255.255.0.0
  mtu 1500
  atm route-bridged ip
  pvc 100/100
   encapsulation aal5snap
!
```

Precedence-Based Aggregate WRED Configuration Example

The following example shows a precedence-based aggregate WRED configuration:

```
! Create a policy map named prec-aggr-wred.
!
Router(config)# policy-map prec-aggr-wred
!
! Configure a default class for the policy map.
!
Router(config-pmap)# class class-default
!
! Enable precedence-based (the default setting) aggregate WRED for the default class.
!
Router(config-pmap-c)# random-detect aggregate
!
! Define an aggregate subclass for packets with IP Precedence values of 0-3 and assign the
! WRED profile parameter values for this subclass.
!
Router(config-pmap-c)# random-detect precedence values 0 1 2 3 minimum thresh 10
maximum-thresh 100 mark-prob 10
!
! Define an aggregate subclass for packets with IP Precedence values of 4 and 5 and assign
! the WRED profile parameter values for this subclass.
!
Router(config-pmap-c)# random-detect precedence values 4 5 minimum-thresh 40
maximum-thresh 400 mark-prob 10
!
! Define an aggregate subclass for packets with an IP Precedence value of 6 and assign the
! WRED profile parameter values for this subclass.
!
Router(config-pmap-c)# random-detect precedence values 6 minimum-thresh 60 maximum-thresh
600 mark-prob 10
```

```
!
! Define an aggregate subclass for packets with an IP Precedence value of 7 and assign the
! WRED profile parameter values for this subclass.
!
Router(config-pmap-c)# random-detect precedence values 7 minimum-thresh 70 maximum-thresh
700 mark-prob 10
!
! Attach the policy map prec-aggr-wred to the interface. Note all ATM SPA service policies
! are applied at the atm vc level.
!
Router(config-pmap-c)# interface ATM4/1/0.10 point-to-point
Router(config-subif)# ip address 10.0.0.2 255.255.255.0
Router(config-subif)# pvc 10/110
Router(config-subif)# service policy output prec-aggr-wred
```

DSCP-Based Aggregate WRED Configuration Example

The following example shows a DSCP-based aggregate WRED configuration:

```
! Create a policy map named dscp-aggr-wred.
!
Router(config)# policy-map dscp-aggr-wred
!
! Configure a default class for the policy map.
!
Router(config-pmap)# class class-default
!
! Enable dscp-based aggregate WRED for the default class and assign the
! default WRED profile parameter values to be used for all subclasses that have not been
! specifically configured..
!
Router(config-pmap-c)# random-detect dscp-based aggregate minimum-thresh 1 maximum-thresh
10 mark-prob 10
!
! Define an aggregate subclass for packets with DSCP values of 0-7 and assign the WRED
! profile parameter values for this subclass
!
Router(config-pmap-c)# random-detect dscp values 0 1 2 3 4 5 6 7 minimum-thresh 10
maximum-thresh 20 mark-prob 10
!
! Define an aggregate subclass for packets with DSCP values of 8-11 and assign the WRED
! profile parameter values for this subclass.
!
Router(config-pmap-c)random-detect dscp values 8 9 10 11 minimum-thresh 10 maximum-thresh
40 mark-prob 10
!
! Attach the policy map dscp-aggr-wred to the interface. Note all ATM SPA service policies
! are applied at the atm vc level.
!
Router(config)# interface ATM4/1/0.11 point-to-point
Router(config-subif)# ip address 10.0.0.2 255.255.255.0
Router(config-subif) pvc 11/101
Router(config-subif)# service policy output dscp-aggr-wred
```

Switched Virtual Circuits Configuration Example

```
interface ATM4/0/2
 ip address 10.23.33.2 255.255.255.0
 atm clock INTERNAL
 atm pvp 244
 atm esi-address 111111111111.11
 pvc 0/5 qsaal
!
 pvc 0/16 ilmi
!
!
interface ATM4/0/2.1 multipoint
 ip address 10.20.0.2 255.0.0.0
 atm esi-address 333333333333.33
!
 svc nsap 47.0091810000000001011B8C601.222222222222.22
  protocol ip 10.20.0.1
  ubr 1000
!
!
```

```
interface ATM4/0/2.2 multipoint
 ip address 10.13.3.1 255.255.255.0
 atm esi-address 510211111111.11
!
 svc nsap 47.0091810000000001011B8C601.410233333333.33
  protocol ip 10.13.3.3
!
interface ATM4/0/2.3 multipoint
 svc SVC1 nsap 47.009181000000BBBBBB000001.222222222222.22
  protocol ip 33.33.33.1
  broadcast
  encapsulation aal5snap
```

Traffic Parameters for PVCs or SVCs Configuration Example

```
!
interface ATM5/1/1.100 point-to-point
 ip address 10.1.1.1 255.255.255.0
 load-interval 30
 pvc 1/100
  protocol ip 1.1.1.3
  protocol ip 20.1.1.1
  cbr 100
  broadcast
!
!
interface ATM5/1/1.110 point-to-point
 ip address 10.2.2.2 255.255.255.0
 pvc 1/110
  ubr 1000
!
!
interface ATM5/1/1.120 point-to-point
 ip address 10.3.3.3 255.255.255.0
 no ip directed-broadcast
 pvc 1/120
  vbr-nrt 50000 50000
  encapsulation aal5snap
!
!
interface ATM5/1/1.130 point-to-point
 ip address 10.4.4.4 255.255.255.0
 pvc 1/130
  vbr-rt 445 445
  encapsulation aal5snap
!
!
interface ATM5/1/1.140 point-to-point
 ip address 10.5.5.5 255.255.255.0
 atm arp-server nsap 47.00918100000000107B2B4B01.111155550000.00
 atm esi-address 111155550001.00
!
 svc SVC00 nsap 47.00918100000000107B2B4B01.222255550001.00
  protocol ip 10.5.5.6 broadcast
  oam-svc manage
  encapsulation aal5mux ip
 ubr 1000
!
```

Virtual Circuit Classes Configuration Example

```
vc-class atm high-class
  ilmi manage
  oam-pvc manage 5
  oam retry 10 7 3
!
vc-class atm low-class
!
interface ATM4/1/0
 no ip address
 class-int high-class
 atm ilmi-pvc-discovery subinterface
 pvc 0/5 qsaal
!
 pvc 0/16 ilmi
!
!
interface ATM4/1/0.1 multipoint
 pvc 1/110
  protocol 10.10.10.14
!
interface ATM4/1/1
 ip address 10.10.11.2 255.255.255.0
 class-int low-class
 atm uni-version 4.0
 atm pvp 1
 atm esi-address AAAAAAAAAAAA.AA
interface ATM4/1/1.2 multipoint
 pvc 2/100
  protocol ip 10.10.11.1
!
```

Virtual Circuit Bundles Configuration Example

```
!
interface ATM5/1/1
 ip address 1.1.1.1 255.255.255.0
 load-interval 30
 pvc 1/100
  protocol ip 1.1.1.3
  protocol ip 20.1.1.1
  cbr 140000
  broadcast
  oam-pvc manage
!
 pvc 1/101
  protocol ip 9.9.9.2
  encapsulation aal5ciscoppp Virtual-Template1
!
!
interface ATM5/1/1.200 multipoint
 ip address 7.7.7.1 255.255.255.0
 bundle atm-bundle
  pvc-bundle high 2/100
   class-vc high
  pvc-bundle med 2/101
   class-vc med
  pvc-bundle low 2/102
   class-vc low
!
```

Link Fragmentation and Interleaving with Virtual Templates Configuration Example

The following simple example shows a sample LFI configuration using a virtual template interface:

```
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
class-map match-all prec4
  match ip precedence 4
class-map match-all prec5
  match ip precedence 5
class-map match-all prec6
  match ip precedence 6
class-map match-all prec7
  match ip precedence 7
class-map match-all prec0
  match ip precedence 0
class-map match-all prec1
  match ip precedence 1
class-map match-all prec2
  match ip precedence 2
class-map match-all dscp2
  match dscp 2
class-map match-all prec3
  match ip precedence 3
class-map match-all prec8
  match precedence 0  2  4  6
class-map match-any all
class-map match-all any
  match any
!
!
policy-map pmap1
  class prec1
    bandwidth percent 10
  class prec2
     police 100000000 3125000 3125000 conform-action transmit exceed-action drop
    priority
!
!
!
interface ATM2/1/0
 no ip address
 atm clock INTERNAL
!
interface ATM2/1/0.1 point-to-point
 pvc 0/100
  encapsulation aal5snap
  protocol ppp Virtual-Template1
!
!
interface ATM2/1/0.1000 point-to-point
 pvc 1/1000
  encapsulation aal5ciscoppp Virtual-Template2
!
!
interface ATM2/1/0.1001 point-to-point
 pvc 1/1001
  protocol ip 10.10.11.12
  encapsulation aal5ciscoppp Virtual-Template3
```

```
!
interface ATM2/1/1
 no ip address
 shutdown
!
interface ATM2/1/2
 no ip address
 shutdown
!
interface ATM2/1/3
 no ip address
!
interface Virtual-Template1
 bandwidth 100
 ip address 10.34.0.2 255.255.255.0
 no keepalive
 ppp chap hostname north-21
 ppp multilink
 ppp multilink fragment-delay 5
 ppp multilink interleave
 multilink max-fragments 16
 service-policy output pmap1
!
interface Virtual-Template2
 ip address 10.36.0.2 255.255.255.0
 no keepalive
 ppp chap hostname north-22
 ppp multilink
 ppp multilink fragment-delay 5
 ppp multilink interleave
 service-policy output pmap1
!
interface Virtual-Template3
 ppp chap hostname north-23
 ppp multilink
 ppp multilink fragment-delay 5
 ppp multilink interleave
 service-policy output pmap1
!
interface Vlan1
 no ip address
 shutdown
!
```

Distributed Compressed Real-Time Protocol Configuration Example

```
!
interface ATM5/1/0.200 point-to-point
 pvc 10/300
  encapsulation aal5mux ppp Virtual-Template200
!
...
!
interface Virtual-Template200
 bandwidth 2000
 ip address 10.1.200.2 255.255.255.0
 ip rcp header-compression passive
 ip tcp header-compression passive
 ppp chap hostname template200
 ppp multilink
 ppp multilink fragment-delay 8
 ppp multilink interleave
```

```
 ip rtp header-compression passive
 ip tcp compression-connections 64
!
```

Automatic Protection Switching Configuration Example

```
!
interface ATM4/0/0
 description working
 ip address 10.5.5.1 255.255.255.0
 no shutdown
 aps group 1
 aps working 1
 pvc 1/100
  protocol ip 10.5.5.2
!

interface ATM4/0/1
 description protect
 ip address 10.5.5.1 255.255.255.0
 aps group 1
 aps revert 2
 aps protect 0 10.7.7.7
 pvc 1/100
  protocol ip 10.5.5.2
!
interface Loopback1
 ip address 10.7.7.7 255.255.255.0
```

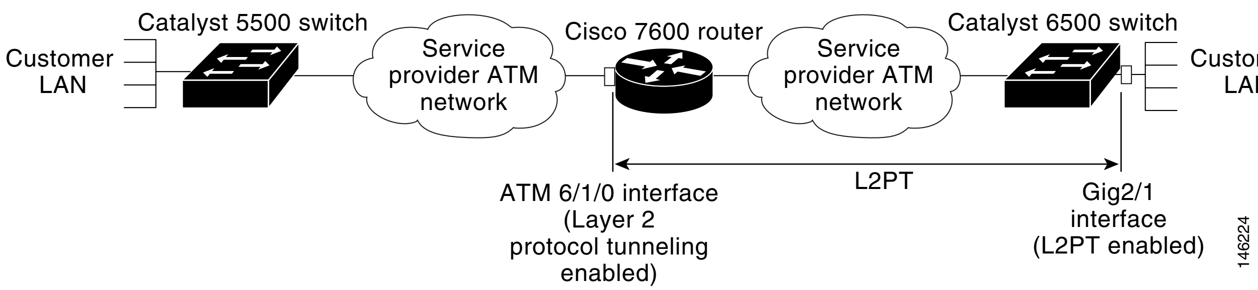
SONET and SDH Framing Configuration Example

```
!
interface ATM2/0/0
 description Example of SONET framing-“atm framing sonet” is default and doesn’t appear
 ip address 10.16.2.2 255.255.255.0
 logging event link-status
 atm sonet report all
 atm sonet threshold sd-ber 3
 atm sonet threshold sf-ber 6
 atm sonet overhead c2 0x00
!
interface ATM2/0/1
 description Example of SDH framing-”atm framing sdh” appears in configuration
 ip address 10.16.3.3 255.255.255.0
 logging event link-status
 atm framing sdh
 atm sonet report all
 atm sonet overhead c2 0x00
!
```

Layer 2 Protocol Tunneling Topology with a Cisco 7600, Catalyst 5500, and Catalyst 6500 Configuration Example

Figure 7-7 shows one sample network topology in which data packets are sent between a Catalyst 6500 series switch and a Cisco 7600 series router.

Figure 7-7 Catalyst 5500 Switch, 6500 Switch, and Cisco 7600 Series Router in an L2PT Topology



As shown in Figure 7-7, Layer 2 Protocol Tunneling (L2PT) is configured at the Cisco 7600 ATM 6/1/0 interface and also at the Catalyst 6500 switch Gig 2/1 interface.

PVST packets are sent from the Catalyst 5500 switch to the Cisco 7600 series router. The Cisco 7600 series router transports those BPDUs by way of L2PT and sends them to the Catalyst 6500 series switch. Those BPDUs are decapsulated and restored before sending the packets out to the customer network.

The Cisco 7600 series router and the Catalyst 6500 series switch are provider edge (PE) devices and the rest are customer edge (CE) devices.

ATM Configuration Example

Any traffic coming in must be sent via a dot1q-tunnel. If the PE VLAN is 200 and the CE VLAN is 100, you have the following configuration:

```
Router(config)# interface atm 6/1/0
Router(config-if)# pvc 6/200
Router(config-if-atm-vc)# bridge-domain 200 dot1q-tunnel ignore-bpdu-pid pvst-tlv 100
```

Ethernet Configuration Example

An example of the Ethernet configuration follows:

```
Router(config)# interface gig2/1
Router(config-if)# switchport
Router(config-if)# switchport access vlan 200
Router(config-if)# switchport mode dot1q-tunnel
Router(config-if)# l2protocol-tunnel
```

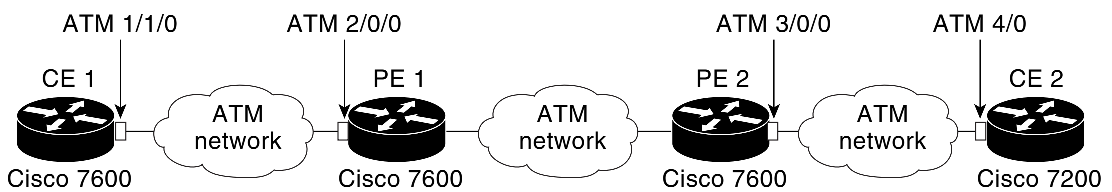
CE VLAN 100 is what is used at the customer sites. The Catalyst 5500 switch sends the IEEE BPDU in data format. The Cisco 7600 series router receives the BPDU and first converts it to PVST+ format. Then the destination address (DA) MAC of the frame is changed to the protocol tunnel MAC address and sent out into the Layer 2 cloud.

At the other end, when the frame leaves the Gig 2/1 interface, the DA MAC is changed back to the PVST+ DA MAC and the PVST+ BPDU is sent to the customer premises equipment (CPE) device.

Layer 2 Protocol Tunneling Topology with a Cisco 7600 and Cisco 7200 Configuration Example

Figure 7-8 shows how a Cisco 7600 series router needs to communicate with a Cisco 7200 series router.

Figure 7-8 Cisco 7600 and Cisco 7200 Routers in an L2PT Topology



PE Configuration

On the PE routers, the configuration appears as follows:

```
!On PE 1
interface ATM2/0/0
    no ip address
    atm mtu-reject-call
    pvc 7/101
    bridge-domain 200 dot1q-tunnel
    !
end
!On PE 2
interface ATM3/0/0
    no ip address
    pvc 2/101
    bridge-domain 200 dot1q-tunnel pvst-tlv 100
    !
end
```

Cisco 7600 CE Configuration

The configuration for the Cisco 7600 CE 1 router would be as follows:

```
!On CE 1
interface ATM1/1/0
    no ip address
    atm mtu-reject-call
    pvc 7/101
    bridge-domain 101
    !
end
```

Cisco 7200 CE Configuration

The configuration for the Cisco 7200 CE 2 router would be as follows:

```
!On CE 2
interface ATM4/0
    no ip address
    no atm ilmi-keepalive
    pvc 2/101
    !
    bridge-group 101
end
```

Data Transmission Sequence from the Cisco 7200 CE to the Cisco 7600 CE

Given the configurations and topologies shown in these examples, the data transmission sequence from the Cisco 7200 CE to the Cisco 7600 CE is as follows:

1. The Cisco 7200 CE 2 router sends BPDUs without the MAC header in RFC 1483 format.
2. The Cisco 7600 PE router receives the packets and then translates the IEEE BPDU into PVST+ BPDU format.
3. VLAN 100 is inserted into the PVST+ BPDU.
4. The frame's destination address (DA) MAC value is rewritten to use the protocol tunnel DA MAC and is sent out into the ATM network cloud.
5. The L2PT BPDU must go out of the PE 1 ATM 2/0/0 interface. The DA MAC is restored to the PVST+ DA MAC.
6. Finally, the PVST+ BPDU is sent to the Cisco 7600 CE 1 router.

Cisco 7600 Basic Back-to-Back Scenario Configuration Example

Figure 7-9 shows an example of a basic back-to-back scenario.

Figure 7-9 Cisco 7600 Routers in Basic Back-to-Back Topology

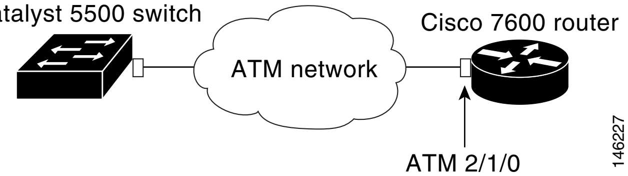
The PDUs exchanged are PVST+ BPDUs. The PVST+ BPDUs are sent using a PID of 0x00-07. The configuration is set as follows:

```
Router(config)# interface atm 2/1/0
Router(config-if)# pvc 2/202
Router(config-if-atm-vc)# bridge-domain 101
```

Catalyst 5500 Switch and Cisco 7600 Series Routers in Back-to-Back Topology Configuration Example

Figure 7-10 shows another sample topology with a simple back-to-back setup, which serves to test basic Catalyst 5500 and Cisco 7600 interoperability.

Figure 7-10 Catalyst 5500 Switch and Cisco 7600 Routers in Back-to-Back Topology



When connected to a device that sends and receives IEEE BPDUs in data format (PID 0x00-07) such as the Catalyst 5000's ATM module, the configuration must be something like this:

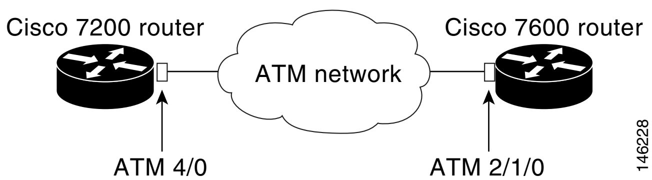
```
Router(config)# interface atm 2/1/0
Router(config-if)# pvc 2/202
Router(config-if-atm-vc)# bridge-domain 101 ignore-bpdu-pid pvst-tlv 101
```

The Cisco 7600 series router translates its outgoing PVST+ BPDUs into IEEE BPDUs. Because the **ignore-bpdu-pid** keyword is also enabled, the BPDU uses a PID of 0x00-07, which is exactly what the Catalyst 5500 switch requires.

Cisco 7600 and Cisco 7200 in Back-to-Back Topology Configuration Example

When connecting to a device that is completely RFC 1483-compliant, in which the IEEE BPDUs are sent using a PID of 0x00-0E, you must use the new **ignore-bpdu-pid** keyword in the **bridge-domain** command. Figure 7-11 shows an example of such a configuration.

Figure 7-11 Cisco 7600 Router Series and Cisco 7200 Router Series in Back-to-Back Topology



For example, when a Cisco 7600 series router is connected to a Cisco 7200 series router, the configuration would be as follows:

```
Router(config)# interface atm 2/1/0
Router(config-if)# pvc 2/202
Router(config-if-atm-vc)# bridge-domain 101 pvst-tlv 101
```

Note In this configuration scenario, the CE's VLAN number must be identical to the **bridge-domain** VLAN number.

An example of the Ethernet configuration is shown in the "Ethernet Configuration Example" section on page 7-80.

CHAPTER 8

Troubleshooting the ATM SPAs

This chapter describes how to monitor and troubleshoot the asynchronous transfer mode (ATM) shared port adapters (SPAs) in a Cisco 7600 series router. This document covers the 1-Port OC-48c/STM-16 ATM SPA, 1-Port OC-12c/STM-4 ATM SPA, and the 2-Port and 4-Port OC-3c/STM-1 ATM SPA.

For more information about troubleshooting your hardware installation, refer to the Cisco 7600 Series Router SIP, SSC, and SPA Hardware Installation Guide.

General Troubleshooting Information

This section provides the following general information for troubleshooting ATM SPA cards and their SPA interface processor (SIP) carrier cards:

Interpreting Console Error and System Messages

To view the explanations and recommended actions for Cisco 7600 series router error messages, including messages related to Cisco 7600 series router SIPs and SPAs, refer to the *Cisco 7600 Series Cisco IOS System Message Guide, Cisco IOS Release 12.2 SX*.

System error messages are organized in the documentation according to the particular system facility that produces the messages. The SIP and SPA error messages use the following facility names:

- Cisco 7600 SIP-200
- Cisco 7600 SIP-400
- 1-Port OC-12c/STM-4 ATM SPA
- 1-Port OC-48c/STM-16 ATM SPA
- 2-Port and 4-Port OC-3c/STM-1 ATM SPA

Using debug Commands

Along with the other debug commands supported on the Cisco 7600 series router, you can obtain specific debug information for SPAs on the Cisco 7600 series router using the **debug hw-module subslot** privileged exec command.

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead can affect system use.

The **debug hw-module subslot** command is intended for use by Cisco Systems technical support personnel. For more information about the debug hw-module subslot command and about other debug commands that can be used on a Cisco 7600 series router, refer to the *Cisco 7600 Series Cisco IOS Command Reference, 12.2 SX* and to the *Cisco IOS Debug Command Reference, Release 12.2 SR*.

Using show Commands

There are several **show** commands that you can use to monitor and troubleshoot the SIP and SPA cards on a Cisco 7600 series router. For more information on these commands, see the "Monitoring the ATM SPA" section on page 8-2.

Also see the following chapters in this guide for additional information about these **show** commands:

- Chapter 7, "Configuring the ATM SPAs"

Monitoring the ATM SPA

This section contains the following subsections that describe commands that can be used to display information about the ATM SPA hardware, interfaces, PVCs, SVCs, and APS configuration:

Note The outputs in this document are samples only. The actual output that appears on your router depends on the model of router, type of cards that are installed, and their configuration.

Displaying Hardware Information

Use the following commands to display different types of hardware and system information:

- **show version**—Displaying System Information, page 8-3
- **show hw-module subslot fpd** and **show idprom module**—Displaying Information About the ATM SPA Hardware Revision Levels, page 8-3

- **show controllers atm**—Displaying Information About the ATM Controller Hardware, page 8-4
- **show diag**—Displaying Information About ATM Ports, page 8-5

Displaying System Information

To display information about the router, its system hardware and software, and the number of each type of interface that is installed, use the **show version** command. The following sample output shows a Cisco 7606 router that has two four-port OC-3c ATM SPA cards installed in a Cisco 7600 SIP-400 carrier card, along with a number of Gigabit Ethernet interfaces:

```
Router# show version

Cisco Internetwork Operating System Software
IOS (tm) c6sup2_rp Software (c6sup2_rp-JSV-M), Released Version 12.2(XX) [BLD-sipedon2
187]
Copyright (c) 1986-2004 by cisco Systems, Inc.
Compiled Tue 16-Mar-04 05:13 by jrstu
Image text-base: 0x40020F94, data-base: 0x424B0000

ROM: System Bootstrap, Version 12.2(14r)S1, RELEASE SOFTWARE (fc1)

sup2_7606 uptime is 44 minutes
Time since sup2_7606 switched to active is 43 minutes
System returned to ROM by power-on (SP by power-on)
System image file is "disk0:c6k222-jsv-mz_022204"

cisco CISCO7606 (R7000) processor (revision 1.0) with 458752K/65536K bytes of memory.
Processor board ID TBM06402027
SR71000 CPU at 600Mhz, Implementation 0x504, Rev 1.2, 512KB L2, 2048KB L3 Cache
Last reset from power-on
Bridging software.
X.25 software, Version 3.0.0.
SuperLAT software (copyright 1990 by Meridian Technology Corp).
TN3270 Emulation software.
1 FlexWAN controller (2 ATM).
2 SIP-400 controllers (7 ATM).
1 Dual-port OC12c ATM controller (2 ATM).
1 Virtual Ethernet/IEEE 802.3 interface(s)
8 Gigabit Ethernet/IEEE 802.3 interface(s)
11 ATM network interface(s)
1917K bytes of non-volatile configuration memory.
8192K bytes of packet buffer memory.
65536K bytes of Flash internal SIMM (Sector size 512K).
Configuration register is 0x2102
```

Displaying Information About the ATM SPA Hardware Revision Levels

To display information about the hardware revision of the SPA, as well as the version of the field-programmable device (FPD) that is onboard the SPA, use the **show hw-module subslot fpd** command. Cisco technical engineers might need this information to debug or troubleshoot problems with a SPA installation.

```
Router# show hw-module subslot fpd

==== ====================== ====== =============================================
                             H/W   Field Programmable   Current   Min. Required
Slot Card Type               Ver.  Device: "ID-Name"    Version      Version
==== ====================== ====== ================== =========== ==============
 5/0 4xOC-3 ATM SPA          1.0   1-I/O FPGA             0.70        0.70
---- ---------------------- ------ ------------------ ----------- --------------
```

```
 5/1 4xOC-3 ATM SPA          1.0   1-I/O FPGA             0.70        0.70
==== ===================== ====== ============================================
```

In addition, the **show idprom module** command also displays the serial number and board revisions for the ATM SPA.

```
Router# show idprom module 5/2

IDPROM for SPA module #5/2
        (FRU is '4-port OC3/STM1 ATM Shared Port Adapter')
        Product Identifier (PID) : SPA-4XOC3-ATM
        Version Identifier (VID) : V01
        PCB Serial Number        : PRTA0304088
        Top Assy. Part Number    : 68-2177-01
        73/68 Board Revision     : 04
        73/68 Board Revision     : 10
        Hardware Revision        : 0.17
        CLEI Code                : UNASSIGNED
```

Displaying Information About the ATM Controller Hardware

To display information about the controller hardware for an ATM interface, including framing and alarm configuration, as well as port, packet, and channel performance statistics, use the **show controllers atm** command, which has the following syntax:

show controllers atm *slot*/*sublot*/*port*

The following example shows typical output for an ATM SPA interface:

```
Router# show controllers atm 5/1/0

Interface ATM5/1/0 is up
 Framing mode: SONET OC3 STS-3c

SONET Subblock:
SECTION
  LOF = 0          LOS    = 0                            BIP(B1) = 603
LINE
  AIS = 0          RDI    = 2          FEBE = 2332       BIP(B2) = 1018
PATH
  AIS = 0          RDI    = 1          FEBE = 28         BIP(B3) = 228
  LOP = 0          NEWPTR = 0          PSE  = 1          NSE     = 2

Active Defects: None
Active Alarms:  None
Alarm reporting enabled for: SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA

ATM framing errors:
  HCS (correctable):   0
  HCS (uncorrectable): 0

APS

  COAPS = 0          PSBF = 0
  State: PSBF_state = False
  Rx(K1/K2): 00/00  Tx(K1/K2): 00/00
  Rx Synchronization Status S1 = 00
  S1S0 = 00, C2 = 00

PATH TRACE BUFFER : STABLE

BER thresholds:  SF = 10e-3  SD = 10e-6
```

```
TCA thresholds:  B1 = 10e-6   B2 = 10e-6   B3 = 10e-6

  Clock source:  line
```

Note The ATM SPA does not support automatic updates of the remote host information, if any, in the Path Trace Buffer section of the **show controllers atm** command.

Displaying Information About ATM Ports

To display information about the type of port adapters that are installed in the router, use the **show diag** command, which has the following syntax:

show diag *slot*

where *slot* is the slot number that contains the port adapter. The following example shows typical output for a 4-port OC-3c ATM SPA that is in slot 4 in the router:

```
Router# show diag 4

Slot 4: Logical_index 8
        4-adapter SIP-200 controller
        Board is analyzed ipc ready
        HW rev 0.300, board revision 08
        Serial Number:  Part number: 73-8272-03

        Slot database information:
        Flags: 0x2004   Insertion time: 0x1961C (01:16:54 ago)

        Controller Memory Size:
                384 MBytes CPU Memory
                128 MBytes Packet Memory
                512 MBytes Total on Board SDRAM
        IOS (tm) cwlc Software (sip1-DW-M), Released Version 12.2(17)SX [BLD-sipedon2 107]

        SPA Information:
        subslot 4/0: 4xOC-3 ATM SPA (0x3E1), status: ok
        subslot 4/1: 4xOC-3 ATM SPA (0x3E1), status: ok
```

Displaying Information About ATM Interfaces

Use the following commands to display information about ATM interfaces:

- **show interface atm**—Displaying Layer 2 Information About an ATM Interface, page 8-5
- **show atm interface atm**—Displaying ATM-Specific Information About an ATM Interface, page 8-6
- **show ip interface**—Displaying Layer 3 IP Information About an ATM Interface, page 8-7

Displaying Layer 2 Information About an ATM Interface

To display Layer 2 information about an ATM interface or subinterface, along with the current status and packet counters, use the **show interface atm** command. The following example shows sample output for an ATM interface on an ATM SPA:

```
Router# show interface atm 5/1/0
```

```
ATM5/1/0 is up, line protocol is up
  Hardware is ATM SPA, address is 000a.f330.2a80 (bia 000a.f330.2a80)
  MTU 4470 bytes, sub MTU 4470, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ATM, loopback not set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 21 current VCCs
  VC idle disconnect time: 300 seconds
  Signalling vc = 1, vpi = 0, vci = 5
        UNI Version = 4.0, Link Side = user
  6 carrier transitions
  Last input 01:47:05, output 00:00:01, output hang never
  Last clearing of "show interface" counters 01:03:35
  Input queue: 0/75/33439/80 (size/max/drops/flushes); Total output drops: 963306
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     9502306 packets input, 6654982829 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     45011 input errors, 131042 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     27827569 packets output, 21072150159 bytes, 0 underruns
     0 output errors, 0 collisions, 3 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

The following example shows sample output for a subinterface on this same ATM interface:

```
Router# show interface atm 5/1/0.200

ATM5/1/0.200 is up, line protocol is up
  Hardware is ATM SPA, address is 000a.f330.2a80 (bia 000a.f330.2a80)
  Internet address is 10.10.10.16/24
  MTU 4470 bytes, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  NSAP address: 47.00918100000000107B2B4B01.222255550001.00
  Encapsulation ATM
  12630 packets input, 10521156 bytes
  4994 packets output, 4176213 bytes
  3753 OAM cells input, 4366 OAM cells output
  AAL5 CRC errors : 0
  AAL5 SAR Timeouts : 0
  AAL5 Oversized SDUs : 0
```

Note The value for “packets output” in the default version of the **show interfaces atm** command includes the bytes used for ATM AAL5 padding, trailer and ATM cell header. To see the packet count without the padding, header, and trailer information, use the **show interfaces atm statistics** or **show atm pvc** commands.

Displaying ATM-Specific Information About an ATM Interface

To display Layer 2 ATM-specific information about an ATM interface or subinterface, use the **show atm interface atm** command:

```
Router# show atm interface atm 3/1/0

Interface ATM3/1/0:
AAL enabled:  AAL5  , Maximum VCs: 1023, Current VCCs: 1

Maximum Transmit Channels: 64
```

```
Max. Datagram Size: 4528
PLIM Type: SONET - 155000Kbps, TX clocking: LINE
Cell-payload scrambling: ON
sts-stream scrambling: ON
0 input, 0 output, 0 IN fast, 0 OUT fast, 0 out drop
 Avail bw = 155000
Config. is ACTIVE
```

Displaying Layer 3 IP Information About an ATM Interface

To display Layer 3 (IP-layer) information about an ATM interface, use the **show ip interface** command. To display a brief summary about all interfaces, use the following command:

show ip interface brief

To display information about a specific ATM interface, use the following command:

show ip interface atm *slot*/*subslot*/*port*

The following output shows a typical example for the brief version of the **show ip interface** command:

```
Router# show ip interface brief

Interface                  IP-Address      OK? Method Status                Protocol
Vlan1                      unassigned      YES NVRAM  down                  down
GigabitEthernet1/1         172.18.76.57    YES NVRAM  up                    up
GigabitEthernet1/2         unassigned      YES NVRAM  administratively down down
ATM3/0/0                   unassigned      YES manual up                    up
ATM3/0/0.1                 unassigned      YES manual up                    up
ATM3/0/0.2                 10.1.1.1        YES manual up                    up
ATM3/1/0                   unassigned      YES manual up                    up
ATM3/1/0.1                 unassigned      YES manual up                    up
ATM3/1/0.2                 unassigned      YES unset  up                    up
ATM3/1/0.3                 11.1.1.1        YES manual up                    up
```

Displaying Information About PVCs and SVCs

Use the following commands to display information about PVCs and SVCs, including mapping, traffic, and VLAN configuration information:

- **show atm vp**—Displaying Information About Virtual Paths, page 8-8
- **show atm vc**—Displaying Information About Virtual Channels, page 8-8
- **show atm pvc**—Displaying Information About PVCs, page 8-9
- **show atm svc** and **show atm ilmi-status**—Displaying Information About SVCs, page 8-10
- **show atm map**—Displaying Information About Layer 2/Layer 3 Mappings, page 8-11
- **show atm traffic**—Displaying Information About ATM Traffic, page 8-12
- **show atm vlan**—Displaying Information About VLAN Mappings, page 8-12
- **show atm class-links**—Displaying Information About VC Bundles, page 8-13

Displaying Information About Virtual Paths

To display information about the virtual paths (VPs) that are configured on the router's ATM interfaces, use the **show atm vp** command:

```
Router# show atm vp

                     Data  CES     Peak    CES
Interface     VPI    VCs   VCs     Kbps    Kbps      Status
ATM5/0/3      1      1     0       149760  0         ACTIVE
ATM5/0/3      1      2     0       299520  299000    ACTIVE
ATM5/0/3      2      0     0       1000    0         ACTIVE

Router#
```

To display detailed information about a specific virtual path, including its current PVCs and SVCs, specify the VPI with the **show atm vp** command:

```
Router# show atm vp 30

ATM8/1/0 VPI: 30,
ATM8/1/0 VPI: 30, PeakRate: 149760, CesRate: 0, DataVCs: 1, CesVCs: 0, Status: ACTIVE

  VCD    VCI   Type   InPkts   OutPkts   AAL/Encap     Status
  2      3     PVC    0        0         F4 OAM        ACTIVE
  3      4     PVC    0        0         F4 OAM        ACTIVE
  4      300   PVC    5        5         AAL5-SNAP     ACTIVE
  6      11    PVC    12       1         AAL5-SNAP     ACTIVE

TotalInPkts: 17, TotalOutPkts: 6, TotalInFast: 0, TotalOutFast: 6, TotalBroadcasts: 0
TotalInPktDrops: 0, TotalOutPktDrops: 0
```

Displaying Information About Virtual Channels

To display information about all of the virtual channels that are currently configured on the ATM interfaces, use the **show atm vc** command without any options:

```
Router# show atm vc

           VCD /                                        Peak  Avg/Min Burst
Interface  Name         VPI   VCI  Type   Encaps   SC   Kbps   Kbps   Cells  Sts
3/0/0      1              1   100  PVC    SNAP     UBR  149760                UP
3/0/1      1              2   100  PVC    SNAP     UBR  149760                UP
3/0/2      1              3   100  PVC    SNAP     UBR  149760                UP
3/0/2      2              3   300  PVC    SNAP     UBR  149760                UP
3/0/3      1              4   100  PVC    SNAP     UBR  149760                UP
```

To display detailed information about a specific virtual connection, specify its VC descriptor (VCD) along with the command:

```
Router# show atm vc 20

   ATM1/1/0.200: VCD: 20, VPI: 2, VCI: 200
   UBR, PeakRate: 44209
   AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
   OAM frequency: 0 second(s)
   InARP frequency: 5 minutes(s)
   Transmit priority 4
   InPkts: 10, OutPkts: 11, InBytes: 680, OutBytes: 708
   InPRoc: 10, OutPRoc: 5, Broadcasts: 0
   InFast: 0, OutFast: 0, InAS: 0, OutAS: 6
   InPktDrops: 0, OutPktDrops: 0
```

```
   CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0
   OAM cells received: 0
   OAM cells sent: 0
   Status: UP
```

You can also display information about the VCs on a specific ATM interface and its subinterfaces:

```
Router# show atm vc interface atm 2/1/0

ATM2/0.101: VCD: 201, VPI: 20, VCI: 101
UBR, PeakRate: 149760
AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 0 second(s)
InARP frequency: 15 minutes(s)
Transmit priority 4
InPkts: 3153520, OutPkts: 277787, InBytes: 402748610, OutBytes: 191349235
InPRoc: 0, OutPRoc: 0, Broadcasts: 0
InFast: 211151, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0, OutPktDrops: 17
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0
OAM cells received: 0
OAM cells sent: 0
Status: UP
```

To display information about the traffic over a particular VC, use the **show atm vc** command with the following syntax:

show atm vc traffic interface atm *slot*/*subslot*/*port vpi vci*

```
Router# show atm vc traffic interface atm 1/0/1 1 101

Interface         VPI  VCI   Type      rx-cell-cnts     tx-cell-cnts
ATM1/0/1          1    101   PVC            9345              7231
```

Displaying Information About PVCs

Use the **show atm pvc** command to provide information about the PVCs that are currently configured on the router. To display all PVCs that are currently configured on the router's ATM interfaces and subinterfaces, use the **show atm pvc** command:

```
Router# show atm pvc

           VCD /                                         Peak  Avg/Min Burst
Interface  Name         VPI   VCI  Type   Encaps    SC   Kbps   Kbps   Cells  Sts
2/1/0      1              2    32  PVC    SNAP      UBR     0                   UP
2/1/0.1    0              0    33  PVC    MUX       UBR 599040                  UP
2/1/0.2    2              0    34  PVC    MUX       UBR 599040                INAC
2/1/0.3    3              0    35  PVC    MUX       UBR 599040                INAC
2/1/0.4    4              0    36  PVC    MUX       UBR 599040                INAC
2/1/1.1    0              0    33  PVC    MUX       UBR 599040                  UP
2/1/1.2    2              0    34  PVC    MUX       UBR 599040                INAC
2/1/1.3    3              0    35  PVC    MUX       UBR 599040                INAC
2/1/1.4    4              0    36  PVC    MUX       UBR 599040                INAC
```

Tip To display all PVCs on a particular ATM interface or subinterface, use the **show atm pvc interface atm** command.

To display detailed information about a particular PVC, specify its VPI/VCI values:

```
Router# show atm pvc 1/100
```

```
ATM3/0/0: VCD: 1, VPI: 1, VCI: 100
UBR, PeakRate: 149760
AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 0 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Disabled
OAM VC status: Not Managed
ILMI VC status: Not Managed
InARP frequency: 15 minutes(s)
Transmit priority 6
InPkts: 94964567, OutPkts: 95069747, InBytes: 833119350, OutBytes: 838799016
InPRoc: 1, OutPRoc: 1, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 94964566, OutAS: 95069746
InPktDrops: 0,  OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0, LengthViolation: 0, CPIErrors: 0
Out CLP=1 Pkts: 0
OAM cells received: 0
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
OAM cells sent: 0
F5 OutEndloop: 0, F5 OutSegloop: 0, F5 OutRDI: 0
F4 OutEndloop: 0, F4 OutSegloop: 0, F4 OutRDI: 0
OAM cell drops: 0
Status: UP

VC 1/100 doesn't exist on 7 of 8 ATM interface(s)
```

Displaying Information About SVCs

Use the **show atm vc** and **show atm ilmi-status** commands to provide information about the SVCs that are currently configured on the router. To display all SVCs that are currently configured on the router's ATM interfaces and subinterfaces, use the **show atm svc** command:

```
Router# show atm svc

           VCD /                                          Peak  Avg/Min Burst
Interface  Name         VPI   VCI  Type   Encaps    SC    Kbps   Kbps   Cells  Sts
4/0/0      1              0     5  SVC    SAAL      UBR  155000                 UP
4/0/2      4              0    35  SVC    SNAP      UBR  155000                 UP
4/1/0      16             0    47  SVC    SNAP      UBR  155000                 UP
4/1/0.1    593            0    80  SVC    SNAP      UBR  599040                 UP
```

Tip To display all SVCs on a particular ATM interface or subinterface, use the **show atm svc interface atm** command.

To display detailed information about a particular SVC, specify its VPI/VCI values:

```
Router# show atm svc 0/35

ATM5/1/0.200: VCD: 3384, VPI: 0, VCI: 35, Connection Name: SVC00
UBR, PeakRate: 155000
AAL5-MUX, etype:0x800, Flags: 0x44, VCmode: 0x0
OAM frequency: 10 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Received
OAM VC status: Verified
ILMI VC status: Not Managed
VC is managed by OAM.
InARP DISABLED
Transmit priority 6
InPkts: 0, OutPkts: 4, InBytes: 0, OutBytes: 400
```

```
InPRoc: 0, OutPRoc: 4, Broadcasts: 0
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0,  OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OverSizedSDUs: 0, LengthViolation: 0, CPIErrors: 0
Out CLP=1 Pkts: 0
OAM cells received: 10
F5 InEndloop: 10, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
OAM cells sent: 10
F5 OutEndloop: 10, F5 OutSegloop: 0, F5 OutRDI: 0
F4 OutEndloop: 0, F4 OutSegloop: 0, F4 OutRDI: 0
OAM cell drops: 0
Status: UP
TTL: 4
interface =  ATM5/1/0.200, call locally initiated, call reference = 8094273
vcnum = 3384, vpi = 0, vci = 35, state = Active(U10)
, point-to-point call
Retry count: Current = 0
timer currently inactive, timer value = 00:00:00
Remote Atm Nsap address: 47.00918100000000107B2B4B01.111155550001.00
, VC owner: ATM_OWNER_SMAP
```

To display information about the ILMI status and NSAP addresses being used for the SVCs on an ATM interface, use the **show atm ilmi-status** command:

```
Router# show atm ilmi-status atm 4/1/0

Interface : ATM4/1/0 Interface Type : Private UNI (User-side)
ILMI VCC : (0, 16) ILMI Keepalive : Enabled/Up (5 Sec 4 Retries)
ILMI State:       UpAndNormal
Peer IP Addr:     10.10.13.1       Peer IF Name:     ATM 3/0/3
Peer MaxVPIbits:  8               Peer MaxVCIbits:  14
Active Prefix(s) :
47.0091.8100.0000.0010.11b8.c601
End-System Registered Address(s) :
47.0091.8100.0000.0010.11b8.c601.2222.2222.2222.22(Confirmed)
47.0091.8100.0000.0010.11b8.c601.aaaa.aaaa.aaaa.aa(Confirmed)
```

Tip To display information about the SVC signaling PVC and ILMI PVC, use the **show atm pvc 0/5** and **show atm pvc 0/16** commands.

Displaying Information About Layer 2/Layer 3 Mappings

To display the mapping between the mappings between virtual circuits and Layer 3 IP addresses, use the **show atm map** command:

```
Router# show atm map

Map list ATM3/1/0.100_ATM_INARP : DYNAMIC
ip 10.11.11.2 maps to VC 19, VPI 2, VCI 100, ATM3/1/0.100
ip 10.11.11.1 maps to VC 4, VPI 0, VCI 60, ATM3/1/0.102
ip 10.11.13.4 maps to VC 1, VPI 5, VCI 33, ATM3/1/0
ip 10.10.9.20 maps to bundle vc-group1, 0/32, 0/33, 0/34, ATM3/1/0.1, broadcast

Map list ATM3/1/1.200_ATM_INARP : DYNAMIC
ip 10.2.3.2 maps to VC 20, VPI 2, VCI 200, ATM1/1/0.200
ip 10.2.3.10 maps to bundle vc-group2, 0/32, 0/33, 0/34, ATM3/1/1.1, broadcast

Map list ATM4/0/3.95_pvc1 : PERMANENT
ip 10.4.4.4 maps to NSAP CD.CDEF.01.234567.890A.BCDE.F012.3456.7890.1234.12, broadcast,
aal5mux, multipoint connection up, VC 6
```

```
ip 10.4.4.6 maps to NSAP DE.CDEF.01.234567.890A.BCDE.F012.3456.7890.1234.12, broadcast,
aal5mux, connection up, VC 15, multipoint connection up, VC 6
ip 10.4.4.16 maps to VC 1, VPI 13, VCI 95, ATM4/0/3.95, aal5mux
```

Displaying Information About ATM Traffic

To display general information about the traffic over the ATM interfaces, use the **show atm traffic** command:

```
Router# show atm traffic

276875 Input packets
272965 Output packets
2 Broadcast packets
0 Packets received on non-existent VC
6 Packets attempted to send on non-existent VC
272523 OAM cells received
F5 InEndloop: 272523, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
272963 OAM cells sent
F5 OutEndloop: 272963, F5 OutSegloop: 0, F5 OutRDI: 0
0 OAM cell drops
```

To display information about traffic shaping on the ATM interfaces in a particular slot, use the **show atm traffic shaping slot** command:

```
Router# show atm traffic shaping slot 3

Traffic Shaping CAM State : ACTIVE
Shaper Configuration Status :
    Shapers In Use By Config : 3, Shapers Available for Config : 3
Shaper Status in Hardware :
    Shaper 0 : In Use - Port : 0/0/0  Class : best-effort
    Shaper 1 : Not In Use
    Shaper 2 : Not In Use
    Shaper 3 : Not In Use
Statistics :
    Total cell discards : 0, clp0 discards : 0,  clp1 discards : 0
    Free cell buffers : 262143
    Total cells queued : 0
```

Tip You can also use the **show atm vc traffic** command to display traffic information for a particular VC.

Displaying Information About VLAN Mappings

To display the mappings of VLAN IDs to VCs, use the **show atm vlan** command:

```
Router# show atm vlan

        VCD     VLAN-ID
        101     1
        102     2
        103     3
        104     4
        105     5
        106     6
        107     7
        108     8
        109     9
        110     10
```

```
        111     11
        112     12
        113     13
        114     14
        115     15
        116     16
        117     17
        118     18
        119     19
        120     20
        121     21
        122     22
...
        800     11
        801     11
        802     11
        803     11
        804     326
        805     326
        806     326
        807     326
        808     327
        809     327
        810     327
        811     327
```

Tip To display the ports being used by a VLAN, use the **show vlan id** command.

Displaying Information About VC Bundles

To display the relationship between a particular VC and its parent VC class, including the parameters that were inherited from the class and those that were set manually, use the **show atm class-link** command:

```
Router# show atm class-links 0/66

Displaying vc-class inheritance for ATM2/0.3, vc 0/66:
broadcast - VC-class configured on main-interface
encapsulation aal5mux ip - VC-class configured on subinterface
no ilmi manage - Not configured - using default
oam-pvc manage 3 - VC-class configured on vc
oam retry 3 5 1 - Not configured - using default
ubr 10000 - Configured on vc directly
```

Displaying Information About Automatic Protection Switching

When you have configured automatic protection switching (APS) on one or more router, you can show the current APS configuration and status with the **show aps** command, which has the following syntax:

show aps [**atm** *interface* | **controller** | **group** [*number*]]

You can display information about the overall APS configuration and about the specific APS groups that include interfaces that are present in the router.

Displaying the Current APS Status

The **show aps** command, without any options, displays information for all interfaces in the router that are configured as Working or Protect APS interfaces. The following shows sample output for a router with one Working interface and one Protect interface:

```
Router# show aps

ATM4/0/1 APS Group 1: protect channel 0 (inactive)
        bidirectional, revertive (2 min)
        PGP timers (default): hello time=1; hold time=3
        state:
        authentication = (default)
        PGP versions (native/negotiated): 2/2
        SONET framing; SONET APS signalling by default
        Received K1K2: 0x00 0x05
                No Request (Null)
        Transmitted K1K2: 0x20 0x05
                Reverse Request (protect)
        Working channel 1 at 10.10.10.41 Enabled
        Remote APS configuration: (null)

ATM4/0/0 APS Group 1: working channel 1 (active)
        PGP timers (from protect): hello time=3; hold time=6
        state: Enabled
        authentication = (default)
        PGP versions (native/negotiated): 2/2
        SONET framing; SONET APS signalling by default
        Protect at 10.10.10.41
        Remote APS configuration: (null)
```

The following sample output is for the same interfaces, except that the Working interface has gone down and the Protect interface is now active:

```
Router# show aps

ATM4/0/1 APS Group 1: protect channel 0 (active)
        bidirectional, revertive (2 min)
        PGP timers (default): hello time=1; hold time=3
        state:
        authentication = (default)
        PGP versions (native/negotiated): 2/2
        SONET framing; SONET APS signalling by default
        Received K1K2: 0x00 0x05
                No Request (Null)
        Transmitted K1K2: 0xC1 0x05
                Signal Failure - Low Priority (working)
        Working channel 1 at 10.10.10.41 Disabled SF
        Pending local request(s):
                0xC (, channel(s) 1)
        Remote APS configuration: (null)

ATM4/0/0 APS Group 1: working channel 1 (Interface down)
        PGP timers (from protect): hello time=3; hold time=6
        state: Disabled
        authentication = (default)
        PGP versions (native/negotiated): 2/2
        SONET framing; SONET APS signalling by default
        Protect at 10.10.10.41
        Remote APS configuration: (null)
```

Tip To display the same information for a specific ATM interface, use the **show aps atm** *slot/subslot/port* command.

Displaying Information About APS Groups

To display information about the APS groups that are configured on the router, use the **show aps group** command. You can display information for all groups or for a single group. For example, the following example shows a typical display for an individual group:

```
Router# show aps group 2

ATM4/0/0 APS Group 2: working channel 1 (active)
        PGP timers (from protect): hello time=3; hold time=6
        SONET framing; SONET APS signalling by default
        Protect at 10.10.10.7
        Remote APS configuration: (null)

ATM4/0/1 APS Group 2: protect channel 0 (inactive)
        bidirectional, revertive (2 min)
        PGP timers (default): hello time=1; hold time=3
        SONET framing; SONET APS signalling by default
        Received K1K2: 0x00 0x05
                No Request (Null)
        Transmitted K1K2: 0x20 0x05
                Reverse Request (protect)
        Working channel 1 at 10.10.10.7  Enabled
        Remote APS configuration: (null)
```

Note In the above example, both the Working and Protect interfaces in the APS group are on the same router. If the two interfaces are on different routers, the **show aps group** command shows information only for the local interface that is a member of the APS group.

Troubleshooting the ATM Shared Port Adapter

This section describes the following commands and messages that can provide information in troubleshooting the ATM SPA and its interfaces:

- Understanding Line Coding Errors, page 8-16
- Using the Ping Command to Verify Network Connectivity, page 8-16
- Using the Ping Command to Verify Network Connectivity, page 8-16
- Using Loopback Commands, page 8-17
- Using ATM Debug Commands, page 8-26
- Using the Cisco IOS Event Tracer to Troubleshoot Problems, page 8-26

Tip For additional information on troubleshooting specific problems related to PVCs and SVCs, see the TAC tech note web page, at the following URL:

http://www.cisco.com/en/US/tech/tk39/tk48/tech_tech_notes_list.html

Understanding Line Coding Errors

This section provides a brief description of line coding and of the types of errors and alarms that can occur on a line:

- Alarm Indication Signal (AIS)—An AIS alarm indicates that an alarm was raised by a device on a line upstream to the ATM interface. Typically, the device creating the alarm is the adjacent network neighbor, but the AIS signal could also be generated by a device in the service provider's ATM cloud.
- Loss of Frame (LOF)—An LOF alarm occurs when the local interface is using a framing format that does not match the framing format being used on the line. LOF errors could also occur when the line or a device on the line is generating bit errors that are corrupting frames.
- Rx Cell HCS Error (HCSE)—The interface detected an error in the cell's header checksum (HCS) field, which indicates that one or more header bits were corrupted. (This field does not indicate whether any errors occurred in the cell's 48-bit payload.)
- Remote Alarm Indication (RAI) and Far-end Receive Failure (FERF)—An RAI/FERF error indicates that a problem exists between the local ATM interface and the far end, and that the error might not be in the local segment between the local interface and adjacent node.

Using the Ping Command to Verify Network Connectivity

The **ping** command is a convenient way to test the ability of an interface to send and receive packets over the network. The **ping** command sends ICMP echo request packets to a specified destination address, which should send an equal number of ICMP echo reply packets in reply. By measuring the numbering of packets that are successfully returned, as well as how long each packet takes to be returned, you can quickly obtain a rough idea of the Layer 3 to Layer 3 connectivity between two interfaces.

The IP **ping** command has the following syntax:

ping

or

ping *ip-address* [**repeat** *count*] [**data** *hex*] [**size** *datagram-size*]

If you enter just **ping**, the command interactively prompts you for all other parameters. Otherwise, you must specify at least a specific IP address as the destination for the ping. You can also optionally specify the following parameters:

- **repeat** *count*—Number of ICMP echo request packets to send. The default is five packets.
- **data** *hex*—The data pattern, in hexadecimal, to be sent in the ICMP echo request packets.
- **size** *datagram-size*—Specifies the size, in bytes, of the ICMP echo request packets to be sent. The range is 40 to 18024 bytes, with a default of 100 bytes.

Examples

The following shows a typical example of the **ping** command:

```
Router# ping 10.10.10.10

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 10.10.10.10, timeout is 2 seconds:
```

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/15/64 ms
```

Note You must have at least one PVC or SVC defined on an ATM interface before it can respond to an ICMP ping packet.

Using Loopback Commands

The **loopback** commands place an interface in loopback mode, which enables you to use the **ping** command to send packets through the local interface and line, so as to test connectivity. These commands are especially useful when an interface is experiencing a high number of cyclic redundancy check (CRC) errors, so that you can pinpoint where the errors are occurring.

Use the following procedures to perform the different loopback tests:

- Using loopback diagnostic to Create a Local Loopback, page 8-17
- Using loopback line, page 8-22

Tip For more information about using loopbacks to troubleshoot CRC errors on an interface, see the *CRC Troubleshooting Guide for ATM Interfaces* tech note, at the following URL:

http://www.cisco.com/en/US/tech/tk39/tk48/technologies_tech_note09186a00800c93ef.shtml

Using loopback diagnostic to Create a Local Loopback

To perform a local loopback test, in which the transmit data is looped back to the receive data at the physical (PHY) layer, use the **loopback diagnostic** command on an ATM interface. This loopback tests connectivity on the local ATM interface, verifying that the interface's framing circuitry and segmentation and reassembly (SAR) circuitry is operating correctly. This loopback, however, does not test the interface's optics circuitry and ports.

Tip If an ATM interface is currently connected to another ATM interface and passing traffic, shut down the remote ATM interface before giving the **loopback diagnostic** command on the local ATM interface. Otherwise, the remote interface continues to send traffic to the local interface, and the remote network could also start reporting interface and network errors.

Figure 8-1 shows a router-level diagram of a local loopback. Figure 8-2 shows a block-level diagram of a local loopback, as it is performed within the ATM interface circuitry.

Figure 8-1 Performing a Local Loopback—Router Level

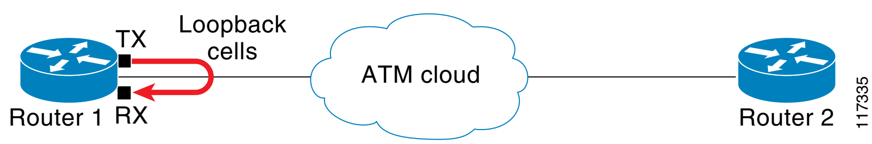
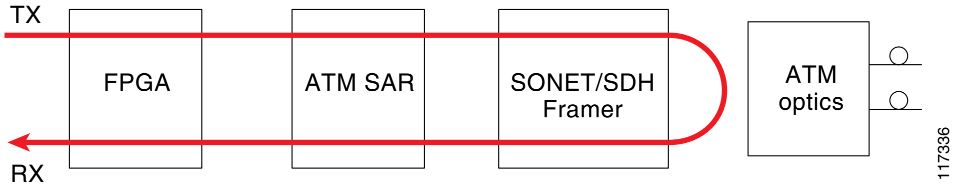


Figure 8-2 Performing a Local Loopback—Block Level



DETAILED STEPS

	Command or Action	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the indicated port on the specified ATM SPA card.
Step 3	Router(config-if)# **loopback diagnostic**	Puts the ATM interface into the local loopback mode, so that data that is transmitted out the interface is internally routed back into the receive data line.
Step 4	Router(config-if)# **atm clock internal**	Specifies that the AMT interface should derive its clocking from its local oscillator, which is required, because the loopback command isolates the interface from the network and from the clocking signals that are derived from the network line.
Step 5	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.
Step 6	Router# **show interface atm** *slot/subslot/port*	(Optional) Verifies that the interface has been configured for loopback mode. The output should show the words "loopback set" when the interface is operating in loopback mode.
Step 7	Router# **debug atm packet interface atm** *slot/subslot/port*	(Optional) Enables packet debugging on the ATM interface. **Note** This command generates several lines of debug output for each packet transmitted and received on the interface. Do not use it on a live network, or you could force the processor to 100% utilization.
Step 8	Router(config-if)# **ping** *ip-address* [**repeat** *count*] [**data** *hex*] [**size** *datagram-size*]	Sends an ICMP echo request packet to the specified IP address. • *ip-address*—Destination IP address for the ICMP echo request packet. Because the interface has been put into loopback mode, the exact IP address does not matter—any valid IP address can be specified. • **repeat** *count*—(Optional) Specifies the number of ICMP echo request packets to be sent. The default is 5. • data hex—(Optional) The data pattern, in hexadecimal, to be sent in the ICMP echo request packets. • **size** *datagram-size*—(Optional) Specifies the size, in bytes, of the ICMP echo request packets to be sent. The range is 40 to 18024 bytes, with a default of 100 bytes. **Note** Because the interface is in loopback mode, the ping command will report that it failed. This is to be expected.

	Command or Action	Purpose

Step 9	Router# **show interface atm** *slot/subslot/port*	Displays interface statistics, including whether any CRC or other errors occurred during the ping test. For example:
		`Router# show interface atm 5/0/1`
		`...`
		`Received 0 broadcasts, 0 runts, 0 giants, 0 throttles`
		`5 input errors, 5 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort`
		`...`
		`Router#`

Step 10	Router(config)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the indicated port on the specified ATM SPA card.
Step 11	Router(config-if)# **no loopback diagnostic**	Removes the local loopback and return the ATM interface to normal operations.

Note Also remember to restore the proper clocking on the local ATM interface and to reenable the remote ATM interface.

Examples

The following sample output shows a local loopback being set with the **loopback diagnostic** command. The **ping** command then sends two PING packets, and the resulting output from the **show interface** command shows that two CRC errors occurred.

```
Router# configure terminal
Router(config)# interface atm 4/1/0
Router(config-if)# loopback diagnostic
Router(config-if)# atm clock internal
Router(config-if)# end
Router# show interface atm 4/1/0

ATM4/1/0 is up, line protocol is up
  Hardware is ATM SPA, address is 000a.f330.2a80 (bia 000a.f330.2a80)
  MTU 4470 bytes, sub MTU 4470, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ATM, loopback set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 21 current VCCs
  VC idle disconnect time: 300 seconds
  Signalling vc = 1, vpi = 0, vci = 5
         UNI Version = 4.0, Link Side = user
  6 carrier transitions
  Last input 01:47:05, output 00:00:01, output hang never
  Last clearing of "show interface" counters 01:03:35
  Input queue: 0/75/33439/80 (size/max/drops/flushes); Total output drops: 963306
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     9502306 packets input, 6654982829 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     27827569 packets output, 21072150159 bytes, 0 underruns
     0 output errors, 0 collisions, 3 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

```
Router# debug atm packet interface atm 4/1/0

ATM packets debugging is on
Displaying packets on interface ATM4/1/0

Router# ping 10.10.10.10 count 2
Type escape sequence to abort.
Sending 2, 100-byte ICMP Echos to 10.10.10.10, timeout is 2 seconds:

1w1d: ATM4/1/0(O):
VCD:0x5 VPI:0x0 VCI:0x55 DM:0x100 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70
1w1d: 4500 0064 001A 0000 FF01 B77A 0101 0102 0101 0101 0800 119A 13A2 07C5 0000
1w1d: 0000 2D41 2408 ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
1w1d: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
1w1d: ABCD ABCD ABCD ABCD ABCD
1w1d:
1w1d: ATM4/1/0(I):
VCD:0x5 VPI:0x0 VCI:0x55 Type:0x0 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70
1w1d: 4500 0064 001A 0000 0101 B57B 0101 0102 0101 0101 0800 119A 13A2 07C5 0000
1w1d: 0000 2D41 2408 ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
1w1d: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
1w1d: ABCD ABCD ABCD ABCD ABCD
1w1d: .
1w1d: ATM4/1/0(O):
VCD:0x5 VPI:0x0 VCI:0x55 DM:0x100 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70
1w1d: 4500 0064 001B 0000 FF01 B779 0101 0102 0101 0101 0800 09C9 13A3 07C5 0000
1w1d: 0000 2D41 2BD8 ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
1w1d: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
1w1d: ABCD ABCD ABCD ABCD ABCD
1w1d:
1w1d: ATM4/1/0(I):
VCD:0x5 VPI:0x0 VCI:0x55 Type:0x0 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70
1w1d: 4500 0064 001B 0000 0101 B57A 0101 0102 0101 0101 0800 09C9 13A3 07C5 0000
1w1d: 0000 2D41 2BD8 ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
1w1d: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
1w1d: ABCD ABCD ABCD ABCD ABCD
1w1d: .
Success rate is 0 percent (0/2)

Router# configure terminal
Router(config)# interface atm 4/1/0
Router(config-if)# no loopback diagnostic
Router(config-if)# end
Router# show interface atm 4/1/0

ATM4/1/0 is up, line protocol is up
  Hardware is ATM SPA, address is 000a.f330.2a80 (bia 000a.f330.2a80)
  MTU 4470 bytes, sub MTU 4470, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ATM, loopback not set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 21 current VCCs
  VC idle disconnect time: 300 seconds
  Signalling vc = 1, vpi = 0, vci = 5
         UNI Version = 4.0, Link Side = user
  6 carrier transitions
  Last input 01:47:05, output 00:00:01, output hang never
  Last clearing of "show interface" counters 01:03:35
  Input queue: 0/75/33439/80 (size/max/drops/flushes); Total output drops: 963306
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     9502306 packets input, 6654982829 bytes, 0 no buffer
```

```
Received 0 broadcasts (0 IP multicast)
0 runts, 0 giants, 0 throttles
2 input errors, 2 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
27827569 packets output, 21072150159 bytes, 0 underruns
0 output errors, 0 collisions, 3 interface resets
0 output buffer failures, 0 output buffers swapped out
```

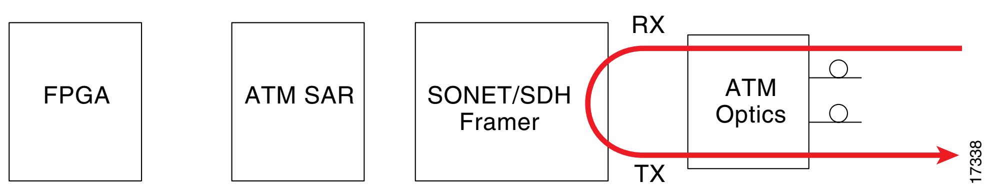
Using loopback line

If an ATM interface can perform a local loopback successfully, without reporting errors, you can next try a line loopback (**loopback line** command) to determine if packet errors are being generated by the ATM network between the local and remote router. In a line loopback, the interface on the remote router is configured with the loopback line command, so that it reflects every packet that it receives back to the originating router. The local router then generates traffic with the ping command to determine whether the line through the network is generating the packet errors.

Figure 8-3 shows a router-level diagram of a line loopback. Figure 8-4 shows a block-level diagram of a line loopback, as it is performed within the ATM interface circuitry.

Figure 8-3 Performing a Local Loopback—Router Level

Figure 8-4 Performing a Line Loopback—Block Level



DETAILED STEPS

	Command or Action	Purpose
Perform the following steps on the remote router:		
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **interface atm** *slot/subslot/port*	Enters interface configuration mode for the indicated port on the specified ATM SPA card.
Step 3	Router(config-if)# **loopback line**	Puts the ATM interface into the line loopback mode, so that it reflects any data it receives back to the originator.
Step 4	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.
Step 5	Router# **show interface atm** *slot/subslot/port*	(Optional) Verifies that the interface has been configured for loopback mode. The output should show the words "loopback set" when the interface is operating in loopback mode.
Perform the following steps on the local router:		
Step 1	Router# **debug atm packet interface atm** *slot/subslot/port*	(Optional) Enables packet debugging on the ATM interface. **Note** This command generates several lines of debug output for each packet transmitted and received on the interface. Do not use it on a live network, or you could force the processor to 100% utilization.
Step 2	Router(config-if)# **ping** *ip-address* [**repeat** *count*] [**data** *hex*] [**size** *datagram-size*]	Sends an ICMP echo request packet to the specified IP address. • *ip-address*—Destination IP address for the ICMP echo request packet. Because the interface has been put into loopback mode, the exact IP address does not matter—any valid IP address can be specified. • **repeat** *count*—(Optional) Specifies the number of ICMP echo request packets to be sent. The default is 5. • data hex—(Optional) The data pattern, in hexadecimal, to be sent in the ICMP echo request packets. The default is 0x0000. • **size** *datagram-size*—(Optional) Specifies the size, in bytes, of the ICMP echo request packets to be sent. The range is 40 to 18024 bytes, with a default of 100 bytes. **Note** Because the interface is in loopback mode, the ping command will report that it failed. This is to be expected.
Step 3	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose

Step 4	Router# **show interface atm** *slot/subslot/port*	Displays interface statistics, including whether any CRC or other errors during the ping test. For example:
		`Router# show interface atm 5/0/1`
		`...`
		`Received 0 broadcasts, 0 runts, 0 giants, 0 throttles`
		`5 input errors, 5 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort`
		`...`
		`Router#`

Note Also remember to remove the loopback mode on the remote ATM interface, using the **no loopback line** command.

Examples

The following shows typical output when performing a line loopback. The following is the output on the remote router:

```
Router# configure terminal
Router(config)# interface atm 3/1/2
Router(config)# loopback line
Router(config)# end
Router# show interface atm 3/1/2

ATM3/1/2 is up, line protocol is up
  Hardware is ATM SPA, address is 000a.330e.2b08 (bia 000a.330e.2b08)
  MTU 4470 bytes, sub MTU 4470, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ATM, loopback set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 103 current VCCs
  VC idle disconnect time: 300 seconds
  Signalling vc = 1, vpi = 0, vci = 5
         UNI Version = 4.0, Link Side = user
  6 carrier transitions
  Last input 00:00:02, output 00:00:01, output hang never
  Last clearing of "show interface" counters 01:03:35
  Input queue: 0/75/13/80 (size/max/drops/flushes); Total output drops: 37
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     932603 packets input, 6798282 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     387275 packets output, 371031501 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

On the Local Router

Perform the following on the local router:

```
Router# debug atm packet interface atm 4/0/0
ATM packets debugging is on
Displaying packets on interface ATM4/0/0
```

```
Router# ping 192.168.100.13 repeat 2 size 128

Type escape sequence to abort.
Sending 2, 128-byte ICMP Echos to 192.168.100.13, timeout is 2 seconds:
..
Success rate is 0 percent (0/2)

00:52:00: ATM4/0/0(O):
VCD:0x1 VPI:0x0 VCI:0x55 DM:0x100 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70
00:52:00: 4500 0064 000F 0000 FF01 B785 0101 0102 0101 0101 0800 CE44 121D 0009 0000
00:52:00: 0000 002F 9DB0 ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD
00:52:00:
00:52:00: ATM4/0/0(I):
VCD:0x1 VPI:0x0 VCI:0x55 Type:0x0 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70
00:52:00: 4500 0064 000F 0000 0101 B586 0101 0102 0101 0101 0800 CE44 121D 0009 0000
00:52:00: 0000 002F 9DB0 ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD
00:52:00:
00:52:02: ATM4/0/0(O):
VCD:0x1 VPI:0x0 VCI:0x55 DM:0x100 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70
00:52:02: 4500 0064 0010 0000 FF01 B784 0101 0102 0101 0101 0800 C673 121E 0009 0000
00:52:02: 0000 002F A580 ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:02: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD
00:52:02:
00:52:02: ATM4/0/0(I):
VCD:0x1 VPI:0x0 VCI:0x55 Type:0x0 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70
00:52:02: 4500 0064 0010 0000 0101 B585 0101 0102 0101 0101 0800 C673 121E 0009 0000
00:52:02: 0000 002F A580 ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:02: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD ABCD
00:52:00: ABCD ABCD ABCD ABCD

Router# show interface atm 4/0/0

ATM4/0/0 is up, line protocol is up
  Hardware is ATM SPA, address is 000a.12f0.80b1 (bia 000a.12f0.80b1)
  MTU 4470 bytes, sub MTU 4470, BW 149760 Kbit, DLY 80 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ATM, loopback not set
  Encapsulation(s): AAL5
  4095 maximum active VCs, 103 current VCCs
  VC idle disconnect time: 300 seconds
  Signalling vc = 1, vpi = 0, vci = 5
         UNI Version = 4.0, Link Side = user
  6 carrier transitions
  Last input 00:00:02, output 00:00:01, output hang never
  Last clearing of "show interface" counters 01:03:35
  Input queue: 0/75/13/80 (size/max/drops/flushes); Total output drops: 37
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
    5 minute input rate 0 bits/sec, 0 packets/sec
    5 minute output rate 0 bits/sec, 0 packets/sec
    94917 packets input, 1638383 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
```

```
 0 input errors, 2 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
 102898 packets output, 2042785 bytes, 0 underruns
0 output errors, 0 collisions, 5 interface resets
0 ouput buffer failures, 0 output buffers swapped out
```

Using ATM Debug Commands

The following debug commands can be useful when troubleshooting problems on an ATM interface or subinterface:

- **debug atm bundle errors—**Displays information about VC bundle errors.
- **debug atm bundle events—**Displays information about events related to the configuration and operation of VC bundles, such as VC bumping, when bundles are brought up, when they are taken down, and so forth.
- **debug atm errors**—Displays errors that occur on an ATM interface, such as encapsulation and framing errors, as well as any errors that might occur during configuration of the ATM interfaces.
- **debug atm events**—Displays information about events that occur on the ATM interfaces, such as changes to the ATM SPA and ATM interface configuration, card and interface resets, and PVC or SVC creation.

Note The output of **debug atm events** can be extremely verbose and can cause problems if large numbers of ATM VCs are configured. The command should only be used when a few VCs are configured.

- **debug atm oam**—Displays the contents of ATM operation and maintenance (OAM) cells as they arrive from the ATM network.
- **debug atm packet**—Displays a hexadecimal dump of each packet’s SNAP/NLPID/SMDS header, followed by the first 40 bytes of the packet.

Tip Use the **no debug all** command to turn off all debugging displays.

For more information about these commands, see the *Cisco IOS Debug Command Reference, Release 12.2*.

Using the Cisco IOS Event Tracer to Troubleshoot Problems

Note This feature is intended for use as a software diagnostic tool and should be configured only under the direction of a Cisco Technical Assistance Center (TAC) representative.

The Event Tracer feature provides a binary trace facility for troubleshooting Cisco IOS software. This feature gives Cisco service representatives additional insight into the operation of the Cisco IOS software and can be useful in helping to diagnose problems in the unlikely event of an operating system malfunction or, in the case of redundant systems, route processor switchover.

Event tracing works by reading informational messages from specific Cisco IOS software subsystem components that have been preprogrammed to work with event tracing, and by logging messages from those components into system memory. Trace messages stored in memory can be displayed on the screen or saved to a file for later analysis.

The SPAs currently support the “spa” component to trace SPA OIR-related events.

For more information about using the Event Tracer feature, refer to the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide09186a0080087164.html

Preparing for Online Insertion and Removal of a SPA

The Cisco 7600 series router supports online insertion and removal (OIR) of the SIP, in addition to each of the SPAs. Therefore, you can remove a SIP with its SPAs still intact, or you can remove a SPA independently from the SIP, leaving the SIP installed in the router.

This means that a SIP can remain installed in the router with one SPA remaining active, while you remove another SPA from one of the SIP subslots. If you are not planning to immediately replace a SPA into the SIP, then be sure to install a blank filler plate in the subslot. The SIP should always be fully installed with either functional SPAs or blank filler plates.

For more information about activating and deactivating SPAs in preparation for OIR, see the “Preparing for Online Insertion and Removal of SIPs and SPAs” topic in the “Troubleshooting a SIP” chapter in this guide.

PART 4

CEoP Shared Port Adapters

CHAPTER 9

Overview of the CEoP and Channelized ATM SPAs

This chapter provides an overview of the release history, features, and MIB support for the Circuit Emulation over Packet (CEoP) shared port adapters (SPAs) that are available for Cisco 7600 series routers. This chapter includes the following sections:

Release History

Release	Modification
12.2(33)SRC	Support was added for the following features: • Support was introduced for the 2-Port Channelized T3/E3 ATM CEoP SPA. • Support was added for Inverse multiplexing over ATM (IMA). • KEOPS Phase 2 Access Circuit Redundancy with Local Switching • KEOPS Phase 2 TDM Local Switching
12.2(33)SRB1	Support was added for the following new features: • ATM pseudowire redundancy. • Out-of-band clocking.
12.2(33)SRB	Support was introduced for the 1-Port Channelized OC-3 STM-1 ATM CEoP SPA and 24-Port Channelized T1/E1 ATM CEoP SPA.

Overview

The CEoP SPAs are single-width, single-height, cross-platform Circuit Emulation over Packet (CEoP) shared port adapters (SPAs) for Cisco 7600 series routers. CEoP SPAs come in the following models:

- 24-Port Channelized T1/E1 ATM CEoP SPA (SPA-24CHT1-CE-ATM=)
- 2-Port Channelized T3/E3 ATM CEoP SPA (SPA-2CHT3-CE-ATM=)
- 1-Port Channelized OC-3 STM-1 ATM CEoP SPA (SPA-1CHOC3-CE-ATM=)

The 24-Port Channelized T1/E1 ATM CEoP SPA and 1-Port Channelized OC-3 STM-1 ATM CEoP SPA must be installed in a Cisco 7600 SIP-400 SPA interface processor (SIP) before they can be used in the Cisco 7600 series router. A maximum of four CEoP SPAs can be installed in each SIP, and these SPAs can be different models. You can install the SPA in the SIP before or after you insert the SIP into the router chassis. This allows you to perform online insertion and removal (OIR) operations either by removing individual SPAs from the SIP, or by removing the entire SIP (and its contained SPAs) from the router chassis.

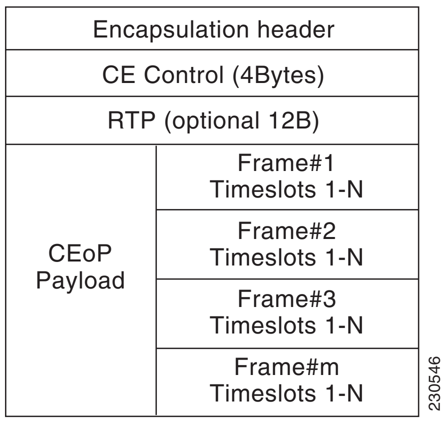
Circuit Emulation over Packet (CEoP) is the imitation of a physical connection. Many service providers and enterprises operate both packet switched networks and time division multiplexed (TDM) networks. These service providers and enterprises have moved many of their data services from the TDM network to their packet network for scalability and efficiency. Cisco provides routing and switching solutions capable of transporting Layer 2 and Layer 3 protocols such as Ethernet, IP, and Frame Relay. While most applications and services have been migrated to the packet-based network, some, including voice and legacy applications, still rely on a circuit or leased line for transport. CEoP SPAs implement Circuit Emulation over Packet by transporting circuits over a packet-based network. CEoP SPAs help service providers and enterprises migrate to one packet network capable of efficiently delivering both data and circuit services. CEoP SPAs also support ATM and ATM pseudowire. For an overview of ATM, see the "ATM Overview" section on page 6-3.

Note In Cisco IOS Release 12.2(33)SRC, the 2-Port Channelized T3/E3 ATM CEoP SPA does not support Circuit Emulation (CEM) mode. The SPA supports ATM mode only.

CEoP Frame Formats

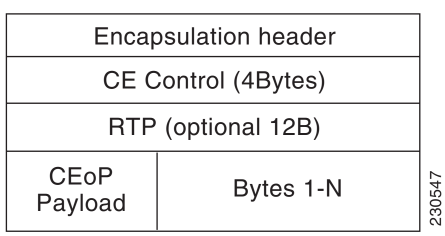
Circuit Emulation Services over Packet Switched Network (CESoPSN) mode is used to encapsulate T1/E1 structured (channelized) services over PSN. Structured mode (CESoPSN) identifies framing and sends only payload, which can be channelized T1s within DS3 and DS0s within T1. DS0s can be bundled to the same packet. Figure 9-1 shows the frame format in CESoPSN mode.

Figure 9-1 Structured Mode Frame Format



Structure-Agnostic TDM over Packet (SAToP) mode is used to encapsulate T1/E1 or T3/E3 unstructured (unchannelized) services over packet switched networks. In unstructured (SAToP) mode, bytes are sent out as they arrive on the TDM line. Bytes do not have to be aligned with any framing. Figure 9-2 shows the frame format in SAToP mode.

Figure 9-2 Unstructured Mode Frame Format



Supported Features

This section provides a list of some of the primary features supported by the CEoP hardware and software:

- Basic Features, page 9-4
- SONET/SDH Error, Alarm, and Performance Monitoring, page 9-5
- Layer 2 Features, page 9-7
- Layer 3 Features, page 9-7
- High Availability Features, page 9-8

Basic Features

- Circuit emulation compliant with IETF standards for CESoPSN and SAToP
- The 24-Port Channelized T1/E1 ATM CEoP SPA supports T1 or E1, which can be channelized to DS0 for circuit emulation (CEM).
- The 2-Port Channelized T3/E3 ATM CEoP SPA is supported in Cisco IOS Release 12.2(33)SRC and later releases.
- The 1-Port Channelized OC-3 STM-1 ATM CEoP SPA supports VT1.5 SONET channelization, and VC-11 and VC-12 SDH channelizations. ATM can be configured on T1s, while CEM can be configured down to DS0.
- Maintenance Digital Link (MDL) and Far End Alarm Control (FEAC) features (T3/E3)
- Facility Data Link (FDL) support (T1/E1)
- Adaptive clock recovery compliant with G.823 and G.824 Traffic interface ITU specification
- Compliant with Y.1411 ATM-MPLS network interworking—cell mode user plane interworking
- Compliant with Y.1413 TDM-MPLS network interworking—user plane interworking
- Compliant with Y.1453 TDM-IP network interworking—user plane interworking
- ATM MPLS encapsulation IETF RFC and drafts
- ATM over channelized T1 lines
- Full channelization down to DS0 (CEM only)
- Simultaneous multiple interface support (for example, ATM and circuit emulation)
- Bellcore GR-253-CORE SONET/SDH compliance (ITU-T G.707, G.783, G.957, G.958)
- Supports both permanent virtual circuits (PVCs) and switched virtual circuits (SVCs)
- The absolute maximum for the sum of VPs at VCs is 2048 per CEoP SPA. Each interface can have a maximum of 2047 VCs with the following recommended limitations:
  - On a Cisco 7600 SIP-400, 8000 PVCs are supported on multipoint subinterfaces.
  - A recommended maximum number of 2048 PVCs on all point-to-point subinterfaces for all CEoP SPAs in a SIP.
  - A recommended maximum number of 16,380 PVCs on all multipoint subinterfaces for all CEoP SPAs in a SIP, and a recommended maximum number of 200 PVCs per each individual multipoint subinterface.
  - A recommended maximum number of 400 SVCs for all CEoP SPAs in a SIP.
  - A recommended maximum number of 1024 PVCs or 400 SVCs using service policies for all CEoP SPAs in a SIP.
- Up to 4096 simultaneous segmentations and reassemblies (SARs) per interface
- Supports a maximum number of 200 PVCs or SVCs using Link Fragmentation and Interleaving (LFI) for all CEoP ATM SPAs (or other ATM modules) in a Cisco 7600 series router
- Up to 1000 maximum virtual templates per router
- ATM adaptation layer 5 (AAL5) for data traffic
- Hardware switching of multicast packets for point-to-point subinterfaces

- The 1-Port Channelized OC-3 STM-1 ATM CEoP SPA uses small form-factor pluggable (SFP) optical transceivers, allowing the same CEoP SPA hardware to support multimode (MM), short reach (SR), intermediate reach (IR1), and long reach (LR1 and LR2) fiber, depending on the capabilities of the SPA.
- ATM section, line, and path alarm indication signal (AIS) cells, including support for F4 and F5 flows, loopback, and remote defect indication (RDI)
- Operation, Administration, and Maintenance (OAM) cells
- Online insertion and removal (OIR) of individual CEoP SPAs from the SIP, as well as OIR of the SIPs with CEoP SPAs installed

Cisco IOS Release 12.2SRC adds support for the following new features:

- 2-Port Channelized T3/E3 ATM CEoP SPA (supports clear-channel T3 ATM mode only)
- Inverse multiplexing over ATM (IMA)
- CEM local switching and local switching redundancy
- ATM cell packing (VC and VP modes)
- ATM local switching and local switching redundancy

SONET/SDH Error, Alarm, and Performance Monitoring

- Fiber removed and reinserted
- Signal failure bit error rate (SF-BER)
- Signal degrade bit error rate (SD-BER)
- Signal label payload construction (C2)
- Path trace byte (J1)
- Section Diagnostics:
  - Loss of signal (SLOS)
  - Loss of frame (SLOF)
  - Error counts for B1
  - Threshold crossing alarms (TCA) for B1 (B1-TCA)
- Line Diagnostics:
  - Line alarm indication signal (LAIS)
  - Line remote defect indication (LRDI)
  - Line remote error indication (LREI)
  - Error counts for B2
  - Threshold crossing alarms for B2 (B2-TCA)
- Path Diagnostics:
  - Path alarm indication signal (PAIS)
  - Path remote defect indication (PRDI)
  - Path remote error indication (PREI)
  - Error counts for B3
  - Threshold crossing alarms for B3 (B3-TCA)

- Loss of pointer (PLOP)
- New pointer events (NEWPTR)
- Positive stuffing event (PSE)
- Negative stuffing event (NSE)

- The following loopback tests are supported:
  - Network (line) loopback
  - Internal (diagnostic) loopback
- Supported SONET/SDH synchronization:
  - Local (internal) timing (for inter-router connections over dark fiber or wave division multiplexing [WDM] equipment)
  - Loop (line) timing (for connecting to SONET/SDH equipment)
  - +/– 4.6 ppm clock accuracy over full operating temperature

T1/E1 Errors and Alarms

The 24-Port Channelized T1/E1 ATM CEoP SPA reports the following types of T1/E1 errors and alarms:

- Cyclic redundancy check (CRC) errors
- Far end block error (FEBE)
- Alarm indication signal (AIS)
- Remote alarm indication (RAI)
- Loss of signal (LOS)
- Out of frame (OOF)
- Failed seconds
- Bursty seconds
- Bipolar violations
- Error events
- Failed signal rate
- Line and Path Diagnostics:
  - Errored Second–Line (ES-L)
  - Severely Errored Second–Line (SES-L)
  - Coding violation–Line (CV-L)
  - Failure Count–Path (FC-P)
  - Errored Second–Path (ES-P)
  - Severely Errored Second–Path (SES-P)
  - Unavailable Seconds–Path (UAS-P)

T3/E3 Errors and Alarms

The 2-Port Channelized T3/E3 ATM CEoP SPA reports the following errors and alarms:

- AIS (Alarm Indication Signal)
- Far end bit error (FEBE)

- Far end receive failure (FERF)
- Frame error
- Out of frame (OOF)
- Path parity error
- Parity bit (P-bit) disagreements
- Receive Alarm Indication Signal (RAIS)
- Yellow alarm bit (X-bits) disagreements

Layer 2 Features

- Supports the following encapsulation types:
  - AAL5SNAP (LLC/SNAP)
  - LLC encapsulated bridged protocol
  - AAL5MUX (VC multiplexing)
  - AAL5CISCOPPP
- Supports the following ATM traffic classes and per-VC traffic shaping modes:
  - Constant bit rate (CBR) with peak rate
  - Unspecified bit rate (UBR) with peak cell rate (PCR)
  - Non-real-time variable bit rate (VBR-nrt)
  - Variable bit rate real-time (VBR-rt)

Note ATM shaping is supported, but class queue-based shaping is not.

- ATM point-to-point and multipoint connections
- Explicit Forward Congestion Indication (EFCI) bit in the ATM cell header
- Integrated Local Management Interface (ILMI) operation, including keepalive, PVC discovery, and address registration and deregistration
- Link Fragmentation and Interleaving (LFI) performed in hardware
- VC–to–VC local switching and cell relay
- VP–to–VP local switching and cell relay
- AToM VP Mode Cell Relay support
- RFC 1755, *ATM Signaling Support for IP over ATM*
- ATM User-Network Interface (UNI) signalling V3.0, V3.1, and V4.0 only
- RFC 2225, *Classical IP and ARP over ATM* (obsoletes RFC 1577)
- Unspecified bit rate plus (UBR+) traffic service class on SVCs and PVCs

Layer 3 Features

- ATM VC Access Trunk Emulation (multi-VLAN to VC)

- ATM over MPLS (AToM) in AAL5 mode (except for AToM cell packing)
- ATM over MPLS (AToM) in AAL5/AAL0 VC mode
- Distributed Link Fragmentation and Interleaving (dLFI) for ATM (dLFI packet counters are supported, but dLFI byte counters are not supported)
- Network-Based Application Recognition (NBAR)
- 2047 is the maximum number of VCs per interface (assuming no VPs). Each VP reduces the total number of VCs per SPA.
- OAM flow connectivity using OAM ping for segment or end-to-end loopback
- Multicast SVCs are supported if there is only one VC on the subinterface
- PVC multicast (Protocol Independent Multicast [PIM] dense and sparse modes)
- Quality of Service (QoS):
  - Policing
  - IP-to-ATM class of service (IP precedence and DSCP)
  - ATM CLP bits matching for ingress and set ATM CLP bits for egress through MQC for PVC
- RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*:
  - PVC bridging (full-bridging)
- Routing protocols:
  - Border Gateway Protocol (BGP)
  - Enhanced Interior Gateway Routing Protocol (EIGRP)
  - Interior Gateway Routing Protocol (IGRP)
  - Integrated Intermediate System-to-Intermediate System (IS-IS)
  - Open Shortest Path First (OSPF)
  - Routing Information Protocol version 1 and version 2 (RIPv1 and RIPv2)

High Availability Features

- 1+1 Automatic Protection Switching (APS) redundancy (PVC circuits only)
- Route Processor Redundancy (RPR)
- RPR Plus (RPR+)
- OSPF Nonstop Forwarding (NSF)

Cisco IOS Release 12.2SRC adds support for the following high-availability feature:

- NonStop Forwarding and Stateful switchover (NSF/SSO) support for CEM and ATM pseudowires

Unsupported Features

- MLPPP and MLFR are not supported
- Primary surge protection for the 24-Port Channelized T1/E1 ATM CEoP SPA
- The following High Availability features are not supported:
  - APS 1:N redundancy is not supported.
  - APS redundancy is not supported on SVCs.
  - APS reflector mode (**aps reflector** interface configuration command) is not supported.
- PVC autoprovisioning (**create on-demand** VC class configuration command) is not supported.
- Creating SVCs with UNI signalling version 4.1 is not supported (UNI signalling v 3.0, v 3.1, and v 4.0 are supported).
- Enhanced Remote Defect Indication–Path (ERDI-P) is not supported.
- Fast Re-Route (FRR) over ATM is not supported.
- LAN Emulation (LANE) is not supported.
- Available Bit Rate (ABR) traffic service class is not supported.
- Oversubscription of the Cisco 7600 SIP-400 is not supported (in either CEM or ATM mode).

Prerequisites

- The Cisco 7600 SIP-400 requires a Cisco 7600 series router using a SUP-720 processor or Route Switch Processor 720 (RSP720) that is running Cisco IOS Release 12.2(33)SRB or later release.
- Before configuring the CEoP SPA, have the following information available:
  - IP addresses for all ports on the new interfaces, including subinterfaces.

Restrictions

- The 1-Port Channelized OC-3 STM-1 ATM CEoP SPA and 24-Port Channelized T1/E1 ATM CEoP SPA do not support mixed line modes (for example, T1 or E1, or T3). A reset of the SPA is required to change modes.
- The 1-Port Channelized OC-3 STM-1 ATM CEoP SPA,the 2-Port Channelized T3/E3 ATM CEoP SPA, and the 24-Port Channelized T1/E1 ATM CEoP SPA do not support the following features: BRE, LFI, RBE, or bridging.
- The 2-Port Channelized T3/E3 ATM CEoP SPA can receive data over distances of up to 1350 ft (411.5 meters).
- When a pseudowire is configured on an interface, APS for the interface is useful only in conjunction with pseudowire redundancy.

Supported MIBs

The following MIBs are supported in Cisco IOS Release 12.2(33)SRB and later releases for the CEoP SPAs on the Cisco 7600 series router.

Common MIBs

- ENTITY-MIB
- IF-MIB
- MIB-II
- MPLS-CEM-MIB

Cisco-Specific MPLS MIBs

- CISCO-IETF-PW-MIB
- CISCO-IETF-PW-MPLS-MIB

Cisco-Specific Common MIBs

- CISCO-ENTITY-EXT-MIB
- OLD-CISCO-CHASSIS-MIB
- CISCO-CLASS-BASED-QOS-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-SENSOR-MIB
- CISCO-MQC-MIB

For more information about MIB support on a Cisco 7600 series router, refer to the *Cisco 7600 Series Internet Router MIB Specifications Guide*, at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/products_mib_quick_reference_book09186a00807f69b0.html

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions at this URL:

http://www.cisco.com/register

Displaying the SPA Hardware Type

To verify the SPA hardware type that is installed in your Cisco 7600 series router, use the **show interfaces** or **show diag** commands. A number of other **show** commands also provide information about the SPA hardware.

Table 9-1 shows the hardware description that appears in the **show** command output for each type of CEoP SPA that is supported on the Cisco 7600 series router:

Table 9-1 CEoP SPA Hardware Descriptions in show Commands

SPA	Description in show interfaces Command
SPA-24CHT1-CE-ATM	"Hardware is SPA-24CHT1-CE-ATM"
SPA-1CHOC3-CE-ATM	"Hardware is SPA-1CHOC3-CE-ATM"
SPA-2CHT3-CE-ATM	"Hardware is SPA-2CHT3-CE-ATM"

Example of the show interfaces cem Command

The following example shows output from the **show interfaces cem** command on a Cisco 7600 series router with an CEoP SPA installed in the first subslot of a SIP that is installed in slot 2:

```
Router# show interfaces cem 2/1/3
CEM2/1/3 is up, line protocol is up
  Hardware is Circuit Emulation Interface
  MTU 1500 bytes, BW 10000000 Kbit, DLY 0 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation CEM, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

CHAPTER 10

Configuring the CEoP and Channelized ATM SPAs

This chapter provides information about configuring the Circuit Emulation over Packet (CEoP) shared port adapters (SPAs) on the Cisco 7600 series router. It contains the following sections:

For information about managing your system images and configuration files, see the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications for your Cisco IOS software release.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Configuration Tasks

This section describes the most common configurations for the CEoP SPAs on a Cisco 7600 series router. It contains procedures for the following:

- Specifying the Interface Address on a SPA, page 10-2
- Configuring Port Usage (Overview), page 10-2

Specifying the Interface Address on a SPA

Four CEoP SPAs can be installed in a SPA interface processor (SIP). Ports are numbered from left to right, beginning with 0. Single-port SPAs use only the port number 0. To configure or monitor SPA interfaces, you need to specify the physical location of the SIP, SPA, and interface in the command-line-interface (CLI). The interface address format is *slot*/*subslot*/*port*, where:

- *slot*—Specifies the chassis slot number in the Cisco 7600 series router where the SIP is installed
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed
- *port*—Specifies the number of the individual interface port on a SPA

The following example shows how to specify the first interface (0) on a SPA installed in subslot 1 of the SIP in chassis slot 3:

```
Router(config)# interface cem 3/1/0
```

For more information about how to identify slots and subslots, see the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section on page 4-2.

Configuring Port Usage (Overview)

The 24-Port Channelized T1/E1 ATM CEoP SPA and 1-Port Channelized OC-3/STM-1 ATM CEoP SPA can be configured to run in the following modes:

- Circuit emulation (CEM)
- Channelized Asynchronous Transfer Mode (ATM)
- Inverse Multiplexing over ATM (IMA)

The 2-Port Channelized T3/E3 ATM CEoP SPA, introduced in Cisco IOS Release 12.2(33)SRC, can be configured to run in ATM mode. The SPA does not currently support CEM or IMA mode.

The following tables show the commands to configure each of the SPAs for CEM or ATM. Detailed configuration instructions are provided in the sections that follow.

Configuring the 24-Port Channelized T1/E1 ATM CEoP SPA

To configure the 24-Port Channelized T1/E1 ATM CEoP SPA, perform the following steps:

	Command or Action	Purpose
Step 1	Router(config)# **card type** {**t1** | **e1**} *slot subslot*	Selects a card type.
Step 2	Router(config)# **controller** {**t1** | **e1**} *slot/subslot/port*	Selects the controller for the SPA port to configure.

	Command or Action	Purpose
Step 3	Router(config-controller)# **cem-group** *group* **unframed**	Creates a SAToP CEM group and configures the port for clear-channel CEM mode.
Step 4	Router(config-controller)# **cem-group** *group* **timeslots** *1-24*	Creates a CESoPSN CEM group and configures the port for channelized CEM mode.
Step 5	Router(config-controller)# **atm**	Configures the port for ATM and creates an ATM interface.
Step 6	Router(config-controller)# **ima-group** *group-number*	Configures the interface to run in IMA mode, and assigns the interface to an IMA group.

Configuring the 2-Port Channelized T3/E3 ATM CEoP SPA

To configure the 2-Port Channelized T3/E3 ATM CEoP SPA, perform the following steps:

	Command or Action	Purpose
Step 1	Router(config)# **controller** {**t3** | **e3**} *slot/subslot/port*	Selects the controller for the SPA port to configure. **Note** For Cisco IOS Release 12.2(33)SRC, you must select **t3**; **e3** is not supported.
Step 2	Router(config-controller)# **atm**	Configures the port to run in clear-channel ATM mode and creates an ATM interface to represent the port.

Note See the "Configuring T3" section on page 10-16 for information about the features that are not supported on the SPA in Cisco IOS Release 12.2SRC.

Configuring the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA for SONET VT1.5

To configure the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA for SONET VT 1.5, perform the following steps:

	Command or Action	Purpose
Step 1	Router(config)# **controller sonet** *5/1/0*	Selects the controller to configure.
Step 2	Router(config-controller)# **framing sonet**	Specifies SONET framing.
Step 3	Router(config-controller)# **sts-1** *2*	Specifies the STS identifier.
Step 4	Router(config-ctrlr-sts1)# **mode** *vt-15*	Specifies VT-15 as the STS-1 mode of operation.
Step 5	Router(config-ctrlr-sts1)# **vtg** *3* **t1** *2* **atm**	Creates a T1 (VT1.5) ATM interface.
Step 6	Router(config-ctrlr-sts1)# **vtg** *1* **t1** *1* **ima-group** *group-number*	Configures the interface to run in IMA mode and assigns the interface to an IMA group.
Step 7	Router(config-ctrlr-sts1)# **vtg** *2* **t1** *1* **cem-group** *1* **unframed**	Creates a single SAToP CEM group.
Step 8	Router(config-ctrlr-sts1)# **vtg** *2* **t1 4 cem-group** *2* **timeslots** *1-5,14*	Creates a CESoPSN CEM group.

Configuring the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA for SDH AU-4 C-12

To configure the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA for SDH AU-4 C-12, perform the following steps:

	Command or Action	Purpose
Step 1	Router(config)# **controller sonet** *5/1/0*	Selects the controller to configure.
Step 2	Router(config-controller)# **framing sdh**	Specifies SDH as the framing mode.
Step 3	Router(config-controller)# **aug mapping au-4**	Specifies AUG mapping.
Step 4	Router(config-controller)# **au-4** *1* **tug-3** *2*	Selects the AU-4, TUG-3 to configure.
Step 5	Router(config-ctrlr-tug3)# **mode c-12**	Specifies the channelization mode for the TUG-3.
Step 6	Router(config-ctrlr-tug3)# **tug-2** *7* **e1** *3* **atm**	Creates an ATM interface.
Step 7	Router(config-ctrlr-tug3)# **tug-2** *1* **e1** *1* **ima-group** *group-number*	Configures the interface to run in IMA mode and assigns the interface to an IMA group.
Step 8	Router(config-ctrlr-tug3)# **tug-2** *1* **e1** *1* **cem-group** *1* **timeslots** *1-31*	Creates a CESoPSN CEM group.

Configuring the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA for SDH AU-3 C-11

To configure the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA for SDH AU-3 C-11, perform the following steps:

	Command or Action	Purpose
Step 1	Router(config)# **controller sonet** *5/1/0*	Selects the controller to configure.
Step 2	Router(config-controller)# **framing sdh**	Specifies the framing mode.
Step 3	Router(config-controller)# **aug mapping au-3**	Specifies AUG mapping.
Step 4	Router(config-controller)# **au-3** *3*	Selects the AU-3 to configure.
Step 5	Router(config-ctrlr-au3)# **mode** *c-11*	Specifies the channelization mode for the link.
Step 6	Router(config-ctrlr-au3)# **tug-2** *7* **t1** *4* **atm**	Creates an ATM interface.
Step 7	Router(config-ctrlr-au3)# **tug-2** *1* **t1** *2* **cem-group** *2015* **timeslots** *1-12*	Creates a CESoPSN CEM group.

Configuring Circuit Emulation

This section provides information about how to configure circuit emulation on a CEoP SPA. Circuit emulation provides a bridge between a time division multiplexed (TDM) network and a packet network (such as Multiprotocol Label Switching [MPLS]). The router encapsulates TDM data in MPLS packets and sends the data over a CEM pseudowire to the remote provider edge (PE) router. Thus, circuit emulation acts like a physical communication link across the packet network.

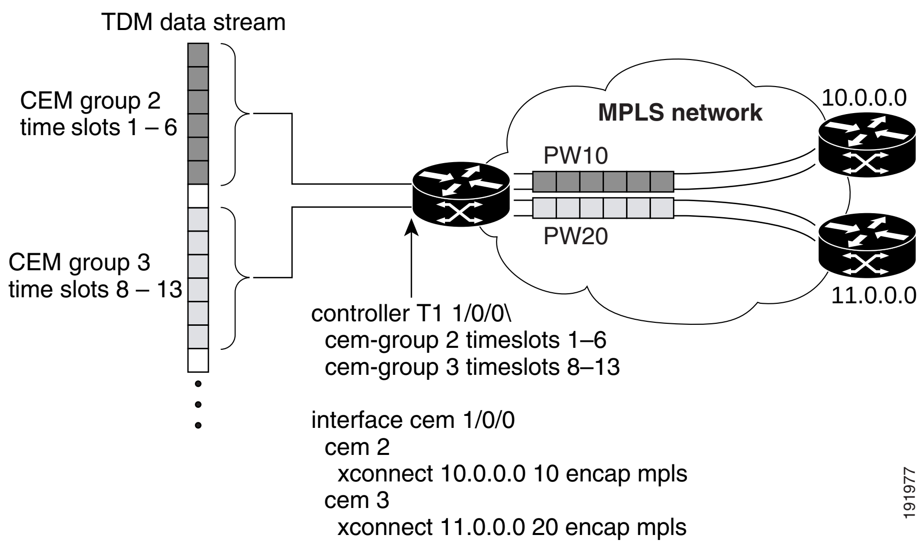
To configure circuit emulation on a CEoP SPA port, you must do the following:

1. Configure one or more CEM groups on the port. Each CEM group represents a set of time slots from the TDM circuit attached to the port. When you configure a CEM group on the port, the router creates an interface that has the same slot/subslot/port number as the port (for example, cem2/1/0).
2. Configure a pseudowire for each CEM group. The router maps the data from the time slots in each group onto its pseudowire and sends the data over the MPLS network to the remote PE router. Use the **xconnect** command with **encap mpls** to create a pseudowire for each CEM group.

Figure 10-1 shows the following sample configuration for a CEoP SPA:

- A TDM circuit is connected to port 0 on a SPA installed in slot 1, subslot 0 (T1 controller 1/0/0).
- Two pseudowires (PW10 and PW20) are configured to carry TDM data across the MPLS network.
- Two CEM groups (2 and 3) are configured for the data in the TDM time slots:
  - Time slots 1 through 6 are sent over pseudowire 10 to the remote PE router at 10.0.0.0.
  - Time slots 8 through 13 are sent to PE router 11.0.0.0 over pseudowire 20.

Figure 10-1 TDM Time Slots to Pseudowire Mappings



Configuring a CEM Group

To configure a CEM group to represent a CEM circuit on a SPA port, use the following procedure.

Note
- The first **cem-group** command under the controller creates a CEM interface that has the same *slot*/*subslot*/*port* information as the controller. The CEM interface is removed when all of the CEM groups under the interface have been deleted.
- The CEM interface is always up, even if the controller state is down. This allows the CEM pseudowire to carry alarm information to the remote end.

	Command or Action	Purpose
Step 1	Router(config)# **controller** *type slot*/*subslot*/*port* Examples Router(config)# **controller t1 3/1/** Router(config)# **controller sonet 2/0/1**	Selects the controller for the port being configured: • *type* identifies the port type. Depending on the card type, valid values are **t1**, **e1**, **t3**, **e3**, or **sonet**. For additional information, see the sections for configuring those port types. • *slot*/*subslot*/*port* identifies the SPA slot, subslot, and port.
Step 2	Router(config-controller)# [**no**] **cem-group** *group-number* {**unframed** | **timeslots** *timeslot*} Examples Router(config)# **controller t1 3/2/0** Router(config-controller)# **cem-group 1 unframed** Router(config)# **controller t1 3/2/1** Router(config-controller)# **cem-group 1 timeslots 1,3,5-11** Router(config-controller)# **cem-group 2 timeslots 12-24**	Creates a CEM circuit (group) from one or more time slots of the line connected to this port. To delete the CEM circuit and release the time slots, use the **no cem-group** *group-number* command. • *group-number* assigns a CEM circuit number: – For T1 controller, the range is 0–23. (24 group IDs) – For E1 controller, the range is 0–30. (31 group IDs) – For T3 controller, the range is 0–671. (672 group IDs) – For E3 controller, the range is 0–479. (480 group IDs) – For OC-3 controller, the range is 0–2015. (2016 group IDs) • **unframed** creates a single CEM circuit from all of the time slots, and uses the framing on the line. Use this keyword for SAToP mode. • **timeslots** *timeslot* specifies the time slots to include in the CEM circuit. Use this keyword for CESoPSN mode. The list of time slots can include commas and hyphens with no spaces between the numbers, commas, and hyphens. **Note** Each time slot operates at 64 kilobits per second (kbps).
Step 3	Router(config-controller)# **exit**	Exits interface configuration mode.

Configuring a CEM Class (Optional)

To assign CEM parameters to one or more CEM interfaces, you can create a CEM class (template) that defines the parameters and then apply the class to the interfaces.

CEM class parameters can be configured directly on the CEM circuit. The inheritance is as follows:

- CEM circuit (highest level)
- Class attached to CEM circuit
- Class attached to the CEM interface

If the same parameter is configured on the CEM interface and CEM circuit, the value on the CEM circuit takes precedence.

To configure a CEM class, use the following procedure:

	Command or Action	Purpose
Step 1	Router(config)# **class cem** *name*	Creates a CEM class to help in configuring parameters in a template and applying parameters at the CEM interface level. • *name* argument is a string of up to 80 characters that identifies the CEM class. Note that the name is truncated to the first 15 characters.
Step 2	Router(config-cem-class)# *command*	Configure CEM parameters by issuing the appropriate commands. See the "Configuring CEM Parameters" section on page 10-41 for commands.

In the following example, a CEM class (TDM-Class-A) is configured to set the payload-size and dejitter-buffer parameters:

```
class cem TDM-Class-A
  payload-size 512
  dejitter-buffer 80
exit
```

In the next example, the CEM parameter settings from TDM-Class-A are applied to CEM interface 2/1/0. Any CEM circuits created under this interface inherit these parameter settings.

```
int cem 2/1/0
  class int TDM-Class-A
  cem 6
      xconnect 10.10.10.10 2 encap mpls
  exit
```

Configuring a CEM Pseudowire

To configure a pseudowire to transport a CEM circuit across the MPLS network, follow this procedure.

	Command or Action	Purpose
Step 1	Router(config)# **interface cem***slot/subslot/port*	Selects the CEM interface where the CEM circuit (group) is located (where *slot/subslot* is the SPA slot and subslot and *port* is the SPA port where the interface exists).
Step 2	Router(config-if)# **cem** *group-number*	Selects the CEM circuit (group) to configure a pseudowire for.
Step 3	Router(config-if-cem)# *command*	(Optional) Defines the operating characteristics for the CEM circuit. For command details, see the "Configuring CEM Parameters" section on page 10-41.
Step 4	Router(config-if)# **xconnect** *peer-router-id vcid* {**encapsulation mpls** | **pseudowire-class** *name*}	Configures a pseudowire to transport TDM data from the CEM circuit across the MPLS network. • *peer-router-id* is the IP address of the remote PE peer router. • *vcid* is a 32-bit identifier to assign to the pseudowire. The same *vcid* must be used for both ends of the pseudowire. • **encapsulation mpls** sets MPLS for tunneling mode. • **pseudowire-class** *name* specifies a pseudowire class that includes the **encapsulation mpls** command. **Note** The *peer-router-id* and *vcid* combination must be unique on the router.
Step 5	Router(config-if)# **exit**	Exits interface configuration mode.

The following sample configuration shows a T1 port on which two CEM circuits (groups) are configured. Each CEM circuit carries data from time slots of the TDM circuit attached to the port.

The two **xconnect** commands create pseudowires to carry the TDM data across the MPLS network. Pseudowire 2 carries the data from time slots 1, 2, 3, 4, 9, and 10 to the remote PE router at 10.10.10.10. Pseudowire 5 carries the data in time slots 5, 6, 7, 8, and 11 to the remote PE router at 10.10.10.11.

```
controller t1 2/1/0
 cem-group 6 timeslots 1-4,9,10
 cem-group 7 timeslots 5-8,11
 framing esf
 linecode b8zs
 clock source adaptive 6
 cablelength long -15db
 crc-threshold 512
 description T1 line to 3rd floor PBX
 loopback network
 no shutdown

int cem2/1/0
  cem 6
    xconnect 10.10.10.10 2 encap mpls
  cem 7
    xconnect 10.10.10.11 5 encap mpls
```

Configuring TDM Local Switching

TDM Local Switching allows switching of Layer 2 data between two CEM interfaces on the same router. The two CEM groups can be on the same physical interface or different physical interfaces; they can be on the same SPA, the same line card, or different line cards.

Note For Cisco IOS Release 12.2(33)SRC, this feature is supported on the 24-Port Channelized T1/E1 ATM CEoP SPA and the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA.

Use the following guidelines for CEoP Phase 2 TDM Local Switching:

- Autoprovisioning is not supported.
- Out-of-band signaling is not supported.
- Port mode local switching is not supported on the CEM interface.
- Interworking with other interface types is not supported.
- The same CEM circuit cannot be used for both local switching and xconnect.
- You can use CEM local switching between two CEM circuits on the same CEM interface.
- CEM local switching can be across a 24-Port Channelized T1/E1 ATM CEoP SPA and a 1-Port Channelized OC-3/STM-1 ATM CEoP SPA.

Use the following procedure to configure CEoPS Phase 2 TDM Local Switching:

	Command or Action	Purpose
Step 1	Router(config)# **interface cem***slot/subslot/port*	Selects the CEM interface to configure the pseudowire for. This is the interface that the TDM circuit is attached to.
Step 2	Router(config)# [**no**] **connect** *name* **cem***x/y/z cemckt1* **cem***a/b/c cemckt2*	Configures a local switching connection between cemckt1 of the CEM interface *x/y/z* and cemckt2 of the CEM interface *a/b/c*. The **no** form of this command unconfigures a local switching connection between cemckt1 of the CEM interface *x/y/z* and cemckt2 of the CEM interface *a/b/c*.

Configuration Example

The following is an example:

```
Router(config)# interface CEM4/3/0
Router(config)# connect cem cem2/1/0 1 cem4/2/0 2
```

Verifying

Use the **show connection**, **show connection al**l, **show connection** id *conn id*, and **show connection** *conn name* commands to verify.

Access Circuit Redundancy with Local Switching

Access Circuit Redundancy with Local Switching provides a backup attachment circuit (AC) when the primary attachment circuit fails. All the ACs must be on same Cisco 7600 series router.

Note For Cisco IOS Release 12.2(33)SRC, this feature is supported on the 24-Port Channelized T1/E1 ATM CEoP SPA and the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA, as well as the 2-Port and 4-Port OC-3c/STM-1 ATM SPA, the 1-Port OC-12c/STM-4 ATM SPA, and the 1-Port OC-48c/STM-16 ATM SPA.

The following combinations of CEM ACs are supported:

- CEM ACs on the same SPA
- CEM ACs on different SPAs on the same SIP
- CEM ACs on different SIPs on the same Cisco 7600 series router

Guidelines

Access Circuit Redundancy with Local Switching guidelines are as follows:

- Autoconfiguration of CEM interfaces is not supported.
- Only the tail end AC can be backed up, if head end fails, there is no protection.
- The circuit type of the primary and backup AC must be identical (failover operation will not switch between different types of interfaces or different CEM circuit types).
- Backs up a local switching connection to cem-ckt3 of CEM interface cem3.Only one backup AC is allowed for each connection.
- Autoconfiguration of backup CEM circuits is not allowed. Autoconfiguration is allowed for backup ATM Permanent Virtual Circuits (PVCs) or ATM Permanent Virtual Paths (PVPs) .
- The CEM circuit used as a backup in a local switching connection cannot be used for xconnect configurations.
- Dynamic modification of parameters in a local switching connection is not supported in the case where the tail-end segment is backed up to a segment using the **backup** command. If you want to modify the parameters in any of the three segments (head-end, tail-end, or backup segment), you must first unconfigure with the **backup** command, make the changes in the individual segments, and then reconfigure the backup with the **backup** command.

Configuration

	Command or Action	Purpose
Step 1	Router(config)# [**no**] **connect** *name* **cem***a/b/c cemckt1* **cem***x/y/z cemckt2*	Configures a local switching connection between cemckt1 of the CEM interface x/y/z and cemckt2 of the CEM interface a/b/c. The **no** form of this command unconfigures a local switching connection between cemckt1 of the CEM interface x/y/z and cemckt2 of the CEM interface a/b/c.
Step 2	Router(config-connection)# **backup interface cem***x/y/z cemckt*	Backs up a locally switched CEM connection.

Configuration Example

The following is a configuration example for Access Circuit Redundancy with Local Switching:

```
Router(config)# connect cem cem2/1/0 1 cem4/2/0 2
Router(config)# backup interface cem 3/0/0 3
```

Verifying

Use the **show xconnect all** command to check the status of the backup and primary circuits.

Configuring ATM

In addition to CEM mode, CEoP SPAs support ATM. When configured to operate in ATM mode, CEoP SPAs support the ATM features listed in Chapter 9, "Overview of the CEoP and Channelized ATM SPAs."

CEoP SPAs also support inverse multiplexing over ATM (IMA), which allows you to combine multiple ATM links into a single high-bandwidth logical link. For more information on IMA, see the "Configuring Inverse Multiplexing over ATM" section on page 10-21.

CEoP SPAs support ATM operation in clear-channel or channelized mode:

- In clear-channel mode, each SPA port provides a single high-speed ATM connection operating at the line rate of the port.
- In channelized mode, each port can be divided into multiple logical channels, each providing a separate ATM connection operating at the channelized line rate (for example, T3 channelized to T1).

Note ATM does not support DS0s. ATM can only be channelized down to T1s.

ATM Connections Per SPA

Use the following guidelines:

- The 24-Port Channelized T1/E1 ATM CEoP SPA provides 24 ATM connections (one for each port) operating at T1 or E1 line rates.

- The 1-Port Channelized OC-3/STM-1 ATM CEoP SPA cannot be configured for clear-channel (OC-3) ATM. Instead, you must channelize the port to T1s or E1s. The number of ATM connections available depends on the configuration mode:
  - Channelized T1 mode provides 84 ATM connections (3 T3 x 28 T1 = 84).
  - Channelized E1 mode provides 63 ATM connections (3 TUG-3/AU-3 x 7 TUG-2 x 3 E1 = 63).
- In clear-channel mode, each port in the 2-Port Channelized T3/E3 ATM CEoP SPA provides a single ATM connection operating at T3 line rate.

ATM Configuration Overview

To configure a port on a CEoP SPA for ATM operation, you must:

1. Set the port to ATM mode. You can also configure IMA (optional).
2. Configure an ATM permanent virtual circuit (PVC) for the port.
3. Configure a pseudowire for the ATM or IMA interface.

ATM and IMA Interfaces

IMA interfaces may consist of groups of T1s or E1s. IMA is not supported on the 2-Port Channelized T3/E3 ATM CEoP SPA.

The router creates an ATM interface for each T3 or E3 port (or channelized T1 or E1) that is configured for ATM mode. The interface has the format **atm***slot*/*subslot*/*port* (where *slot*/*subslot* identifies the SPA slot and subslot and /*port* identifies the port [for example, atm2/1/0]).

If you configure IMA, the router creates an interface to represent each IMA group (link bundle). The interface has the format **atm***slot*/*subslot*/**ima***group-id* (where *slot*/*subslot* identifies the SPA slot and subslot and *group-id* identifies the IMA group number [for example, atm2/1/ima0]).

Configuring an ATM Pseudowire

To configure a pseudowire for an ATM connection or an IMA link bundle, perform these steps. The pseudowire is used to carry the ATM data across the MPLS network.

	Command or Action	Purpose
Step 1	Router(config)# **interface atm***slot*/*subslot*/*port* or Router(config)# **interface atm***slot*/*subslot*/**ima***group-id*	Selects the ATM interface to configure the pseudowire for (where *slot*/*subslot* is the SPA slot and subslot, and /*port* is the SPA port where the interface exists). For IMA, the format is **atm***slot*/*subslot*/**ima***group-id* (where *slot*/*subslot*/ identifies the SPA slot and subslot and *group-id* is the IMA group number).
Step 2	Router(config-if)# **pvc** *vpi*/*vci*	Creates a permanent virtual circuit for the ATM or IMA interface and assigns the PVC a VPI and VCI: • *vpi* specifies the virtual path identifier (0 to 255). • *vci* specifies the virtual channel identifier. Valid values are 32 to 1 less than the value specified by the **atm vc-per-vp** command. **Note** Do not specify 0 for both the VPI and VCI.

	Command or Action	Purpose
Step 3	Router(config-if-atm-vc)# **encapsulation** {**aal0** | **aal5** | **aal5snap**}	Specifies the ATM adaptation layer (AAL) for the PVC: • **aal0**—Selects ATM adaptation layer 0 (cell mode). • **aal5**—Selects ATM adaptation layer 5 (packet mode). • **aal5snap**—Supports Inverse Address Resolution Protocol (ARP). Logical link control/Subnetwork Access Protocol (LLC/SNAP) precedes protocol datagram.
Step 4	Router(config-if-atm-vc)# *command*	Configures the ATM operating characteristics of the PVC. CEoP SPAs support the ATM features in Chapter 9.
Step 5	Router(config-if-atm-vc)# **exit**	Returns you to interface configuration mode.
Step 6	Router(config-if)# **xconnect** *peer-router-id vcid* {**encapsulation mpls** | **pseudowire-class** *name*}	Configures a pseudowire to transport data from the ATM or IMA interface across the MPLS network. • *peer-router-id* is the IP address of the remote PE peer router. • *vcid* is a 32-bit identifier to assign to the pseudowire. The same *vcid* must be used for both ends of the pseudowire. • **encapsulation mpls** sets MPLS for tunneling mode. • **pseudowire-class** *name* specifies a pseudowire class that includes the **encapsulation mpls** command. **Note** The *peer-router-id* and *vcid* combination must be unique on the router.
Step 7	Router(config-if)# **exit**	Exits interface configuration mode.

Configuring Pseudowire Redundancy (Optional)

CEoP SPAs support the L2VPN Pseudowire Redundancy feature, which provides backup service for ATM and circuit emulation (CEM) pseudowires. The L2VPN Pseudowire Redundancy feature enables the network to detect a failure and reroute the Layer 2 (L2) service to another endpoint that can continue to provide service. This feature provides the ability to recover from a failure either of the remote PE router or of the link between the PE and CE routers.

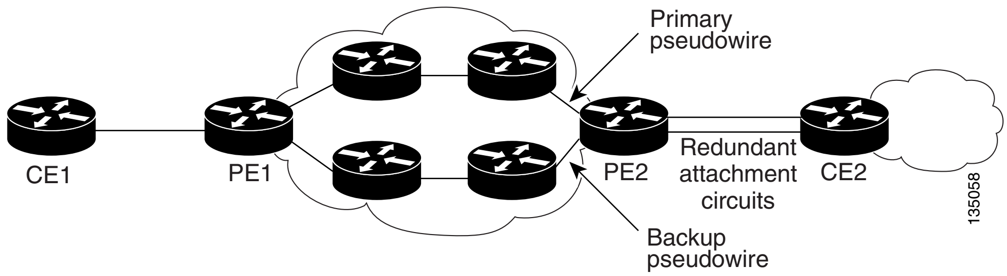
You configure pseudowire redundancy by configuring two pseudowires for an ATM or CEM interface: a primary pseudowire and a backup (standby) pseudowire. If the primary pseudowire goes down, the router uses the backup pseudowire in its place. When the primary pseudowire comes back up, the backup pseudowire is brought down and the router resumes using the primary.

For detailed information about pseudowire redundancy, see its feature description at:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide09186a0080606811.html

Figure 10-2 shows an example of pseudowire redundancy.

Figure 10-2 Pseudowire Redundancy



Following is a summary of the steps to perform to configure pseudowire redundancy on a CEoP SPA. Although an ATM interface is shown, the configuration is the same for CEM.

Note You must configure the backup pseudowire to connect to a different router than the primary pseudowire.

1. **enable**
2. **configure terminal**
3. **interface atm***slot***/***subslot***/***port*
4. **xconnect** *peer-router-id vcid* {**encapsulation mpls** | **pw-class** *pw-class-name*}
5. **backup peer** *peer-router-ip-addr vcid* [**pw-class** *pw-class-name*]
6. **backup delay** *enable-delay* {*disable-delay* | **never**}

The following example shows pseudowire redundancy configured for a CEM circuit (group). In the example, the **xconnect** command configures a primary pseudowire for CEM group 0. The **backup peer** command creates a redundant pseudowire for the group.

```
int cem8/1/1
  no ip address
  cem 0
    xconnect 10.10.10.1 1 encap mpls
    backup peer 10.10.10.2 200
  exit
```

Configuring T1

To configure T1 on a 24-Port Channelized T1/E1 ATM CEoP SPA, use the following procedure and observe these guidelines:

- There can be 0 to 23 channels under a T1 controller, one for each T1 time slot.
- Each channel can be configured as a CEM group.
- Maximum channels under a CEM group is 24.
- Each CEM group number under a controller must be unique.
- A maximum of 191 CEM circuits can be configured.

	Command or Action	Purpose
Step 1	Router(config)# **controller t1** *slot/subslot/port*	Selects the T1 controller.
Step 2	Router(config-controller)# [**no**] **cem-group** *group-number* {**unframed** | **timeslots** *timeslot*}	Creates a CEM interface and assigns it a CEM group number.
Step 3	Router(config-controller)# **framing** {**sf** | **esf**}	Selects the T1 framing type.
Step 4	Router(config-controller)# **exit**	Exits controller configuration mode and returns you to global configuration mode.
Step 5	Router(config)# **interface cem***slot/subslot/port*	Selects the CEM interface.
Step 6	Router(config-controller)# **cem** *group-number*	Selects the specified CEM group.
Step 7	Router(config-controller)# **xconnect** *peer-ip-address* **encap mpls**	Configures a pseudowire for the T1 time slots identified by the CEM group.
Step 8	Router(config-controller)# **exit**	Exits controller configuration mode.

Configuring E1

To configure E1 on a 24-Port Channelized T1/E1 ATM CEoP SPA, use the following procedure:

	Command or Action	Purpose
Step 1	Router(config)# **controller e1** *slot/subslot/port*	Selects the controller for the E1 port being configured.
Step 2	Router(config-controller)# [**no**] **cem-group** *group-number* {**unframed** | **timeslots** *timeslot*}	Creates a CEM interface and assigns a CEM group number.
Step 3	Router(config-controller)# **framing** {**crc4** | **no-crc4**}	Selects the framing type.
Step 4	Router(config-controller)# **exit**	Exits controller configuration mode and returns you to global configuration mode.
Step 5	Router(config)# **interface cem***slot/subslot/port*	Selects the CEM interface.
Step 6	Router(config-controller)# **cem** *group-number*	Selects the specified CEM group.
Step 7	Router(config-controller)# **xconnect** *peer-ip-address* **encap mpls**	Configures a pseudowire for the E1 time slots identified by the CEM group.
Step 8	Router(config-controller)# **exit**	Exits controller configuration mode.

Configuring T3

This section describes how to configure the 2-Port Channelized T3/E3 ATM CEoP SPA. The SPA can be configured to operate in the following modes:

- T3 (clear-channel)
- T1 (channelized T3)
- ATM

In clear-channel mode, each SPA port provides a single high-speed data channel. In channelized mode, each port provides multiple T1/E1 or DS0 channels.

The router creates a logical interface to represent the mode that the SPA port is configured to run in. An ATM interface is created for each T3 port (or channelized T1/E1) that is configured for ATM mode. The interface has the format **atm***slot*/*subslot*/*port* (where *slot*/*subslot* identifies the SPA slot and subslot and /*port* identifies the port). An example is atm2/1/0.

The following sections provide instructions for configuring the SPA:

T3 Configuration Guidelines

This section lists the guidelines for configuring the 2-Port Channelized T3/E3 ATM CEoP SPA. For information about supported features, see Chapter 9, "Overview of the CEoP and Channelized ATM SPAs."

Note For a list of features that are not supported in Cisco IOS Release 12.2SRC, see the "Unsupported Features" section on page 9-9.

T3 Mode

When configured for T3, the SPA can be configured for CEM or ATM clear-channel or channelized mode:

- In clear-channel T3 mode, each SPA port provides a single high-speed data channel operating at 44210 kilobits per second (kbps).
- When configured for CEM, each port supports a maximum of 672 CEM circuits (groups), one for each DS0. Fully channelized, the SPA supports 1344 CEM groups (2 T3s x 28 T1s x 24 DSOs).

ATM Mode

For ATM, note the following guidelines:

- Up to 4000 point-to-point ATM VCs (per SIP) are supported.

Configuring Port Usage

Perform the following steps to configure a SPA port for T3:

Note T3 is not supported with Cisco IOS Release 12.2(33)SRC.

	Command or Action	Purpose
Step 1	Router(config)# **controller** {**t3** | **e3**} *slot*/*subslot*/*port*	Selects the T3 controller for the port you are configuring (where *slot*/*subslot* identifies the SPA slot and subslot and /*port* identifies the port).
Step 2	Router(config-controller)# [**no**] **framing** {**c-bit** | **m13**} Router(config-controller)# [**no**] **framing** {**auto-detect** | **c-bit** | **m23**}	(Optional) Specifies the framing type. The default is C-bit for both clear-channel and channelized modes. For clear-channel mode, configure framing as: • **c-bit**—Specifies C-bit parity framing. • **m13**—Specifies DS3 framing M13 (which is the same as M23). For channelized mode, configure framing as: • **auto-detect**—Detects the framing type at the device at the end of the line and switches to that framing type. If both devices are set to auto-detect, c-bit framing is used. • **c-bit**—Specifies C-bit parity framing. • **m23**—Specifies M23 framing.
Step 3	Router(config-controller)# **clock source** {**internal** | **line**}	(Optional) Specifies the clock source. • **internal**—Selects the internal clock. This is the default for channelized mode. • **line**—Selects the network clock. This is the default for unchannelized mode.
Step 4	Router(config-controller)# **cablelength** *feet*	(Optional) Specifies the length of the cable attached to the port (in feet). Valid values are 0 to 450 ft. The default is 224 ft.
Step 5	Router(config-controller)# [**no**] **loopback** {**local** | **network** | **remote** {**line** | **payload**}}	(Optional) Runs a loopback test, which is useful for troubleshooting problems. The **no** form of the command stops the test. The default is no loopback. • **local**—Loops the signal from Tx to Rx path. Sends alarm indication signal (AIS) to network. • **network**—Loops the signal from Rx to Tx path. • **remote** {**line** | **payload**}—(C-bit framing only) Sends a loopback request to the remote end: **line** loops back the unframed signal and **payload** loops back the framed signal.

	Command or Action	Purpose
Step 6	Router(config-controller)# [**no**] **bert pattern** [**2^11** | **2^15** | **2^20 O.153** | **2^20 QRSS** | **2^23** | **0s** | **1s** | **alt-0-1**] **interval** [*1-1440*]	(Optional) Configures bit-error-rate (BER) testing.
Step 7	Router(config-controller)# **mdl** {**string** {**eic** | **fic** | **generator** | **lic** | **pfi** | **port** | **unit**} *string*} | {**transmit** {**idle-signal** | **path** | **test-signal**}} Example Router(config-controller)# mdl string eic ID Router(config-controller)# mdl string fic Building B Router(config-controller)# mdl string unit ABC Router(config-controller)# mdl string pfi Facility Z Router(config-controller)# mdl string port Port 7 Router(config-controller)# mdl transmit path Router(config-controller)# mdl transmit idle-signal	(Optional) Configures maintenance data link (MDL) messages, which communicate information between local and remote ports. Valid only with C-bit framing. • **mdl string** specifies the type of identification information to include in MDL messages: – **eic** *string* specifies the Equipment Identification Code, up to 10 characters. – **fic** *string* specifies the Frame Identification Code, up to 10 characters. – **generator** *string* specifies the Generator Number for test-signal messages, up to 38 characters. – **lic** *string* is the Location Identification Code, up to 11 characters. – **pfi** *string* specifies the Path Facility Identification Code for path messages, up to 38 characters. – **port** *string* is the port number for idle-signal messages, up to 38 characters. – **unit** *string*—Specifies the Unit Identification Code, up to 6 characters. • **mdl transmit** specifies the type of MDL messages to transmit: – **idle-signal**—Enables idle-signal messages. – **path**—Enables path messages. – **test-signal**—Enables test-signal messages.
Step 8	Router(config-controller)# **exit**	Returns you to global configuration mode.
	If you specified channelized T3 mode in Step 2, see the next section for instructions on how to configure the logical T1/E1 channels on the port. Then, configure the port for CEM or ATM mode ("Configuring Circuit Emulation" or "Configuring the SPA for Clear-Channel ATM").	

Configuring the SPA for Clear-Channel ATM

To configure a T3/E3 SPA port for clear-channel ATM, follow these steps:

	Command or Action	Purpose
Step 1	Router(config)# **controller** {**t3** | **e3**} *slot/subslot/port*	Selects the T3 controller for the port you are configuring (where *slot/subslot* identifies the SPA location and */port* identifies the port).
Step 2	Router(config-controller)# **atm**	Configures the port (interface) for clear-channel ATM. The router creates an ATM interface whose format is atm/*slot/subslot/port* (where *slot/subslot* identifies the SPA slot and subslot and */port* is the SPA port).
Step 3	Router(config-controller)# **exit**	Returns you to global configuration mode.
Step 4	Router(config)# **interface atm***slot/subslot/port*	Selects the ATM interface for the SPA port in Step 1.
Step 5	Router(config-if)# **pvc** *vpi/vci*	Configures a PVC for the interface and assigns the PVC a VPI and VCI. Do not specify 0 for both the VPI and VCI. See the "Configuring an ATM Pseudowire" section on page 10-12 for details on this command and the next.
Step 6	Router(config-if)# **xconnect** *peer-router-id vcid* {**encapsulation mpls** | **pseudowire-class** *name*}	Configures a pseudowire to carry data from the clear-channel ATM interface over the MPLS network.
Step 7	Router(config-if)# **end**	Exits configuration mode.

Configuring SONET (OC-3)

To configure SONET (OC-3) on the1-Port Channelized OC-3/STM-1 ATM CEoP SPA, use the following procedure and observe these guidelines:

- One OC-3 has three SONET paths, each of which can have a T3. Each T3 has 28 T1s.
- Each T3 has a submode for T1 configuration.
- Each T1 can be configured to operate in CEM, ATM, or IMA mode.
- ATM can be configured on T1s only. These modes cannot be configured on T1s that are channelized to DS0s.
- CEM groups can be configured on a T1 directly.
- CEM groups can be channelized to DS0s.
- A maximum of 2016 DS0s can be configured.
- A maximum of 575 CEM circuits can be configured.

To configure the SONET controller, perform this task:

	Command or Action	Purpose
Step 1	Router(config)# **controller sonet** *slot/subslot/port*	Enters the SONET controller configuration submode.
Step 2	Router(config-controller)# **framing sonet**	Configures the controller framing for SONET framing (default).

	Command or Action	Purpose
Step 3	Router(config-controller)# **sts-1** *number*	Specifies the STS identifier.
Step 4	Router(config-ctrlr-sts1)# **mode vt-15**	Specifies VT-15 as the STS-1 mode of operation.
Step 5	Router(config-controller-stsl)# **vtg** *5* **t1** *1* **cem-group** *15* **timeslots** *1-5*,*20-23*	Creates a virtual tributary group carrying a T1.
Step 6	Router(config-controller-stsl)# **exit**	Exits controller configuration mode.

SDH Configuration for AU-4 C-12

This section describes how to enable an interface under SDH framing with AU-4 mapping after configuring the SONET controller.

	Command or Action	Purpose
Step 1	Router(config)# **controller sonet** *5/1/0*	Selects the controller to configure.
Step 2	Router(config-controller)# **framing sdh**	Specifies SDH as the framing mode.
Step 3	Router(config-controller)# **aug mapping au-4**	Specifies AUG mapping.
Step 4	Router(config-controller)# **au-4** *1* **tug-3** *2*	Selects the AU-4, TUG-3 to configure.
Step 5	Router(config-ctrlr-tug3)# **mode c-12**	Specifies the channelization mode for the TUG-3.
Step 6	Router(config-ctrlr-tug3)# **tug-2** *7* **e1** *3* **atm**	Creates an ATM interface.
Step 7	Router(config-ctrlr-tug3)# **tug-2** *1* **e1** *1* **ima-group** *group-number*	Configures the interface to run in IMA mode and assigns the interface to an IMA group.
Step 8	Router(config-ctrlr-tug3)# **tug-2** *1* **e1** *1* **cem-group** *1* **timeslots** *1-31*	Creates a CESoPSN CEM group.

SDH Configuration for AU-3 C-11

This section describes how to enable an interface under SDH framing with AU-3 mapping after configuring the SONET controller.

	Command or Action	Purpose
Step 1	Router(config)# **controller sonet** *5/1/0*	Selects the controller to configure.
Step 2	Router(config-controller)# **framing sdh**	Specifies the framing mode.
Step 3	Router(config-controller)# **aug mapping au-3**	Specifies AUG mapping.
Step 4	Router(config-controller)# **au-3** *3*	Selects the AU-3 to configure.
Step 5	Router(config-ctrlr-au3)# **mode** *c-11*	Specifies the channelization mode for the link.
Step 6	Router(config-ctrlr-au3)# **tug-2** *7* **t1** *4* **atm**	Creates an ATM interface.
Step 7	Router(config-ctrlr-au3)# **tug-2** *1* **t1** *2* **cem-group** *2015* **timeslots** *1-12*	Creates a CESoPSN CEM group.

Configuring Inverse Multiplexing over ATM

Inverse multiplexing over ATM (IMA) allows multiple T1 or E1 links to be bundled together into a high-bandwidth logical link. The rate of the logical IMA link is approximately the sum of the rate of the physical links in the IMA group, although some overhead is required for ATM header and control cells.

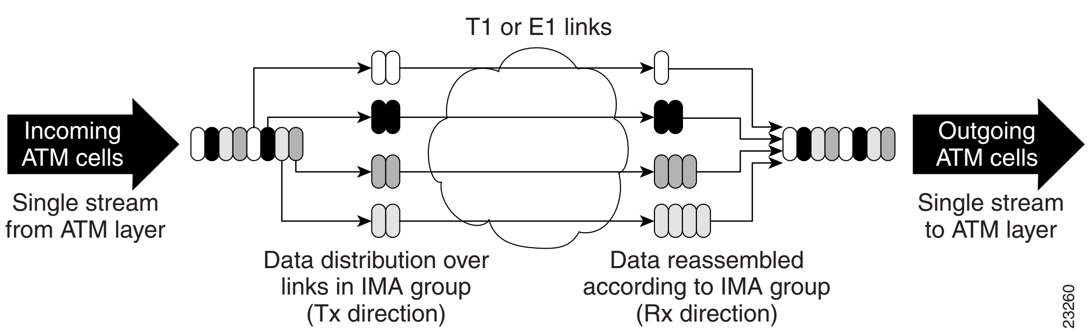
Note IMA is available in Cisco IOS Release 12.2SRC and later releases and is supported on the 24-Port Channelized T1/E1 ATM CEoP SPA and the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA.

The inverse multiplexing operation is transparent to the ATM layer protocols, and therefore the ATM layer can operate normally, as if only a single physical interface is being used. In the transmit direction, IMA takes cells from the ATM layer and sends them in round-robin manner over the individual T1 or E1 links in the IMA group. At the receiving end, the cells are recombined to form the original cell stream and are passed up the ATM layer. An IMA device always sends a continuous stream. If no ATM layer cells are being sent, an IMA filler cell is transmitted to provide a constant stream at the physical layer.

IMA Control Protocol (ICP) cells are periodically transmitted between IMA interfaces. ICP cells control the inverse multiplexing function, provide sequencing for the ATM cell stream, and define the IMA frame. Using an IMA frame length of 128 cells, one out of every 128 cells on each link is an ICP cell.

Figure 10-3 shows how IMA works. In the figure, IMA performs inverse multiplexing and demultiplexing with four bundled links, providing 5.52 Mbps of bandwidth for T1s for packet traffic, after subtracting the overhead of ATM cell headers and ICP cells. The transmitting side, from which cells are distributed across the links, is referred to as Tx, and the receiving side, where cells are recombined, is called Rx.

Figure 10-3 IMA Operation



IMA Configuration Guidelines

Observe these guidelines as you configure the CEoP SPA for inverse multiplexing ATM:

- IMA is supported on the Cisco 7600 SIP-400 with the following CEoP SPAs:
  - 24-Port Channelized T1/E1 ATM CEoP SPA (24 IMA groups per SPA)
  - 1-Port Channelized OC-3/STM-1 ATM CEoP SPA (42 IMA groups per SPA)
- When a T1 or E1 interface is configured for IMA mode, the interface no longer operates as an individual ATM link.
- IMA group numbers (IDs) must be unique on the SPA.

- You cannot mix T1 and E1 lines in the same IMA group.
- The T1 or E1 lines in an IMA group must be on the same CEoP SPA. An IMA group cannot contain T1 or E1 lines from different SPAs.
- Both ends of the T1 or E1 link must be in IMA mode.
- IMA is compliant with nonstop forwarding with stateful switchover (NSF/SSO). This means that when a switchover occurs, IMA connections remain up and continue to pass traffic, with no interruption in service.
- IMA Control Protocol (ICP) cells and filler cells are discarded by the receiving end; therefore, any counters displayed in **show** command output do not include these cells.
- The Cisco 7600 SIP-400 supports a maximum transmission unit (MTU) size of 4470 bytes.

To ensure that IMA groups synchronize correctly after a restart, observe the following guidelines as you configure IMA links. For information about restarts, see the description of **ima autorestart** in the "Configuring IMA Group Parameters" section on page 10-25.

- Each end of an IMA link should have a different IMA group ID. This way, after a restart the router can detect links in loopback mode, which means that a link is communicating with itself instead of the remote end. When both ends of a link have the same group ID, the link is in loopback mode.
- If both ends of an IMA link have the same group ID, loopbacked links might be the first to respond after a restart, in which case the IMA group could be communicating with itself instead of the far end.

IMA Link Bundle Configuration Overview

You bundle T1 or E1 links together by assigning the links to the same IMA group and configuring a PVC for the links in the group to use.

To assign a T1 or E1 link to an IMA group, issue the **ima group** *group-number* command under the T1 or E1 controller for the port that the link is attached to. Bundle a set of links together by issuing **ima group** under the controller for each of the links that you want to add to the bundle, and specify the same group number for each.

The router creates an IMA interface to represent the IMA group (link bundle). The interface has the same slot/subslot information as the SPA, followed by the IMA group ID, as shown here (for example, atm2/1/ima0):

interface atm*slot*/*subslot*/**ima***group-id*

The IMA interface has all of the characteristics of an ATM interface and supports any currently supported ATM features.

When all of the T1/E1 interfaces are removed from an IMA group, the IMA interface that represents the group is removed.

To configure the IMA group for operation, you must:

- Configure a PVC for the links in the IMA group to use.
- Define the operating characteristics of the IMA link bundle by configuring IMA group parameters. (See the "Configuring IMA Group Parameters" section on page 10-25.)

Configuration Example

The following steps provide an example of the steps to configure an IMA link bundle on the 24-Port Channelized T1/E1 ATM CEoP SPA. Detailed steps are provided in the section that follows.

1. Bundle T1 or E1 links together by creating an IMA group and adding each link to the group. In this example, the T1 links attached to ports 0, 1, and 2 of the CEoP SPA in chassis slot 2, SPA subslot 1, are assigned to the same IMA link bundle (IMA group 0). Likewise, the E1 links attached to ports 0 and 1 of the SPA in chassis slot 5, SPA subslot 1 are assigned to another bundle (IMA group 1).

```
controller t1 2/1/0
 ima-group 0
exit
controller t1 2/1/1
 ima-group 0
exit
controller t1 2/1/2
 ima-group 0
exit

controller e1 5/1/0
 ima-group 1
exit
controller e1 5/1/1
 ima-group 1
exit
```

2. Configure a PVC and MPLS pseudowire for the links in the IMA group to use. In the following example, PVC 0/100 is configured for the T1 links in IMA group 0 and PVC 0/101 is configured for the E1 links in IMA group 1:

```
interface  atm2/1/ima0
   pvc 0/100 l2transport
   xconnect 10.2.0.1 10 encapsulation mpls
exit
interface  atm5/1/ima1
   pvc 0/101 l2transport
   xconnect 10.20.0.4 11 encapsulation mpls
exit
```

3. Configure IMA group parameters to define how the links in the group are to operate. In the following example, IMA group 0 is being configured to operate with a minimum of 2 active links, independent clock mode, and a frame length of 256:

```
interface  atm2/1/ima0
   ima active-links-minimum 2
   ima clock-mode independent
   ima frame-length 256
exit
```

Configuring an IMA Link Bundle

To configure an IMA link bundle on a 24-Port Channelized T1/E1 ATM CEoP SPA, perform the following steps from global configuration mode:

	Command or Action	Purpose
Step 1	Router(config)# **controller** {**t1** | **e1**} *slot/subslot/port*	Selects the controller for the link you want to add to an IMA link bundle (an IMA group). • *slot/subslot/port* identifies the chassis slot, SPA subslot, and port being configured.
Step 2	Router(config-controller)# [**no**] **ima-group** *group-number*	Creates an IMA group and adds the link to the group. Use the **no** form of the command remove the link from the IMA group. • *group-number* is a unique ID to assign to the group. Valid values are 0 through 41. **Note** The group number must be unique for the SPA. The 24-Port Channelized T1/E1 ATM CEoP SPA supports 24 IMA groups.
Step 3	Router(config-controller)# **exit**	Returns to global configuration mode.

Repeat steps 1 through 3 to add additional links to the IMA link bundle.

Note All links in an IMA group must be located on the same CEoP SPA.

Step 4	Router(config)# **interface atm***slot/subslot***ima***group-number*	Selects the IMA interface for the link bundle you just created and enters interface configuration mode. • **atm***slot/subslot* specifies the location of the interface. • **ima***group-number* identifies the IMA group.
Step 5	Router(config-if)# **pvc** *vpi/vci*	Configures a PVC for the IMA group and assigns the PVC a VPI and VCI. • *vpi* is the VPI of the PVC. Valid values are 0 to 255. • *vci* is the VCI of the PVC. Valid values are 32 to 1 less than the value set by the **atm vc-per-vp** command. **Note** Do not specify 0 for both the VPI and VCI.
Step 6	Router(config-if)# **xconnect** *peer-router-id vcid* {**encapsulation mpls** | **pseudowire-class** *name*}	Configures a pseudowire to carry data from the IMA link bundle over the MPLS network. See the "Configuring an ATM Pseudowire" section on page 10-12 for details on the command.
Step 7	Router(config-if)# **ima** *command*	Configures parameters for the IMA interface. See Table 10-1 for the configuration commands.
Step 8	Router(config-if)# **end**	Returns you to privileged EXEC mode.

Configuring IMA Group Parameters

Use the commands in Table 10-1 to configure parameters for an IMA group. Issue the commands in interface configuration mode under the IMA interface of the IMA group being configured. Use the **no** form of each command to turn off a feature or to revert to its default setting.

Note If you modify parameters on an IMA interface, the interface is automatically restarted.

Table 10-1 IMA Interface Parameters

Command Name	Description
[**no**] **ima version** {**1.0** | **1.1**}	Selects which version of IMA to use. The default is version 1.1.
[**no**] **ima active-links-minimum** *number*	Specifies the minimum number of IMA links that must be active for the IMA group to be active, where: • *number* is the number of links. Valid values are 1 through 16. The default is 1. The IMA group is active as long as the specified number of links is active; otherwise, the group is brought down and remains out of service until the minimum number of links becomes active again. To determine an appropriate value, consider your application needs and performance requirements, and the number of links in the group.
[**no**] **ima clock-mode** {**common** | **independent**}	Sets the transmit clock mode for the links in the IMA group. The default is common. • **common**—All links use the same clock (which is derived from the specified port). • **independent**—Each link uses a different clock.
[**no**] **ima frame-length** {**32** | **64** | **128** | **256**}	Specifies the number of cells in an IMA frame. The default is 128. Because each IMA frame contains an ICP cell, this parameter also controls how often ICP cells are sent over the links in the IMA group. For example, with a frame length of 64, 1 out of every 64 cells on the link is an ICP cell. The smaller the IMA frame length, the more often ICP cells are sent, which reduces the amount of link bandwidth that is available for data.
[**no**] **ima test** [**link** *link number*] **pattern** *pattern-id*	Sends a continuous test pattern over an IMA link to verify that the link is operational. The pattern is looped back at the receiving end, which is useful for troubleshooting the physical link or configuration problems at the remote end. Use the **no** form of the command to stop the test. • **link** *link number* identifies the IMA link to test. For *link number*, specify the link ID that is displayed by the **show ima interface** *interface* command. Valid values are 0 through 15. • **pattern** *pattern-id* specifies the pattern to use. Valid values are 0 through 255 (0 to 0xFF), although 255 is not recommended. **Note** If you do not specify a link, the test pattern is sent over the first available link.

Table 10-1 IMA Interface Parameters (continued)

Command Name	Description
[**no**] **ima differential-delay-maximum** *milliseconds*	Specifies the maximum allowable differential delay (in milliseconds) among links in the IMA group. If the delay on any link exceeds this value, that link is dropped from the IMA group. IMA sends cells round-robin over the T1 or E1 links in an IMA group, and every link adds some delay. To enable the router to correctly reconstruct the original data stream, IMA adjusts for differences in link delay. However, if a link's delay is greater than the specified maximum, the data stream cannot be reconstructed correctly. Valid values for *milliseconds* are: • 25 to 250 milliseconds (T1) • 25 to 190 milliseconds (E1) A shorter delay allows less adjustment among link delay variations. However, a longer delay can affect overall group performance by adding more latency to traffic or causing retransmission.
[**no**] **ima autorestart** {**near-end-id** *near-end-group-id* [**far-end-id** *far-end-group-id*]}	Enables the auto restart feature, which controls how IMA groups sync up after a restart. The **no** form of the command disables auto restart if it is enabled. See "IMA Auto Restart Examples" for examples. When an IMA group stops operating correctly (for example, due to a failure with the CEoP SPA, an IMA link, or the router), the group must be restarted. When a restart occurs, the local IMA group must sync up with an IMA group at the remote end: • If auto restart is disabled (the default), IMA learns the ID of the remote group each time a restart occurs. In this case, the remote IMA group ID might change between restarts. • If auto restart is enabled, you can specify which remote IMA group the local group should sync up with. This allows you to keep an IMA group from syncing up with any group ID. The **near-end-id** and **far-end-id** keywords identify the IMA groups. Valid values for **near-end-id** is 0-41. Valid values for **far-end-id** are 0-255 . • **near-end-id** *near-end-group-id* is the local IMA group. • **far-end-id** *far-end-group-id* is the remote IMA group. If you specify **near-end-id** only, the local IMA group learns the ID of the remote group to sync up with (which will be the first remote IMA group to become active). This learned remote group ID remains active until the SPA is reloaded. If you specify both **near-end-id** and **far-end-id**, the local IMA group will only synchronize with this remote IMA group. Both the near-end and far-end IDs must be the same.
ima restart	Manually restarts an IMA group. When an IMA group stops operating correctly (for example, due to a link failure), you can use this command to restart the group after the problem has been corrected.

Verifying the IMA Configuration

To display information about all configured IMA groups, or a specific group, use the **show ima interface** command in privileged EXEC mode:

show ima interface atm*slot*/*subslot*/**ima***group-number* [**detail**]

In the following example, information is displayed for IMA group 1 (on the SPA in slot 5, subslot 0):

```
Router# show ima interface atm5/0/ima1
ATM5/0/ima1 is up, ACTIVATION COMPLETE
Slot 5 Slot Unit 0 unit 257, CTRL VC 257, Vir 0, VC -1
IMA Configured BW 12186, Active BW 3046
IMA version 1.0, Frame length 128
Link Test: Disabled
Auto-Restart: Disabled
        ImaGroupState: NearEnd = operational, FarEnd = operational
        ImaGroupFailureStatus = noFailure
IMA Group Current Configuration:
        ImaGroupMinNumTxLinks = 1    ImaGroupMinNumRxLinks = 1
        ImaGroupDiffDelayMax  = 25   ImaGroupNeTxClkMode   = common(ctc)
        ImaGroupFrameLength   = 128  ImaTestProcStatus     = disabled
        ImaGroupTestLink      = None ImaGroupTestPattern   = 0x0
        ImaGroupConfLink      = 8    ImaGroupActiveLink    = 2
IMA Link Information:
 ID       Link              Link Status             Test Status
---- -------------- ------------------------------ ---------------
0    T1 5/0/0       Up   - controller Up           disabled
1    T1 5/0/1       Up   - controller Up           disabled
2    T1 5/0/2       Down - controller Up           disabled
3    T1 5/0/3       Down - controller Up           disabled
4    T1 5/0/4       Down - controller Up           disabled
5    T1 5/0/5       Down - controller Up           disabled
6    T1 5/0/6       Down - controller Up           disabled
7    T1 5/0/7       Down - controller Up           disabled
```

IMA Auto Restart Examples

IMA auto restart is disabled by default, which means that IMA learns the ID of the remote IMA group each time a restart occurs. To see the current settings for auto restart, issue the **show ima interface** command and view the Auto-Restart section of the command output.

Following are several examples of different ways to enable auto restart:

- To enable auto restart so that the local IMA group synchronizes with the first remote IMA group that becomes active, issue the command as follows (where *near-end-group-id* identifies the local IMA group). The learned remote group ID remains active until the SPA is reloaded.

  ima autorestart near-end-id *near-end-group-id*
- To specify which remote IMA group the local IMA group should sync up with, issue the command as follows (where *near-end-group-id* identifies the local IMA group and *far-end-group-id* identifies the remote IMA group). Both near-end and far-end IDs must be the same.

  ima autorestart near-end-id *near-end-group-id* **far-end-id** *far-end-group-id*
- To disable auto restart and have IMA learn the remote IMA group ID after each restart, issue the command as follows:

  no ima autorestart

Configuring Clocking

This section provides information about how to configure clocking on the 24-Port Channelized T1/E1 ATM CEoP SPA and the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA. It describes the following topics:

BITS Clock Support—Receive and Distribute—CEoP SPA on SIP-400

You can use the BITS Clock Support—Receive and Distribute—CEoP SPA on SIP-400 feature to select and configure a clock and distribute it across the chassis to be used as the Transmit reference on all SPA ports.

The BITS Clock support - Receive and Distribute - CEoP SPA on SIP-400 feature is supported on Cisco IOS Release 12.2SRB on the SPA-24CHT1-CE-ATM and the SPA-1CHOC3-CE-ATM, SPA-4XOC3 ATM, SPA-1xOC12/STM4 POS SPAs.

The line card operates in three different modes, dependiing on the configuration and the configured source state.

- Free-running—A line card that is not participating in network-clocking or a line card that is actively sourcing the clock operates in free-running mode. In this mode, the line card internal oscillator generates the reference clock to the backplane.

Note In a nonpartcipating mode or a disabled mode, the linecard distributes a Stratum 3-quality timing signal to an external reference clock. Other interfaces on different linecards receive either the backplane reference clock or the external reference clock depending on their configurations.

- Normal—In normal mode, the module synchronizes with an externally supplied network timing reference, sourced from one of the chassis BITS inputs or recovered from a network interface. In this mode, the accuracy and stability of the output signal is determined by the accuracy and stability of the input reference.

Note Line card operation is in free-running mode only if the SIP-400 is configured as the active sources; otherwise the line cards operate in normal mode.

- Holdover—In holdover mode, the network timing module generates a timing signal based on the stored timing reference used when operating in normal mode. Holdover mode is automatically selected when the recovered reference is lost or has drifted excessivley.

Note You cannot configure the drift range; it is set internally on the line card to +/-9.2 phase shifts per minute (ppm) by default.

Note All linecards operate in the free-running mode until network clock is configured.

Guidelines

Use the following guidelines:

- The SIP-400 operates in free-running mode until network clock is configured.
- When the network clocking configuration is present in the startup configuration, the clocking configuration is not applied until five minutes after the configuration has been parsed. This prevents clocking instability on the backplane when the interfaces/controllers come up out of order.
- Network clocking is enabled by default for the SIP-400.
- Cisco IOS Release 12.2SRB does not support local network clock configurations or synchronization status messaging (SSM).
- If there is a source flap, there is an interval of 180 seconds before the source becomes valid and active.
- In the event of an Out-of-Range (OOR) switchover (revertive mode), the source switchover occurs when the clock offset crosses the -9.2 ppm or +9.2 ppm threshold. If this occurs, you must reconfigure the source.

Configuration Tasks

To configure Network Clocking for the Cisco 7600/SIP-400, use the following commands:

Command or Action	Purpose
Router# **[no] network-clock select** *priority* **interface | controller | slot | system** *interface name* [**global**][**local**]	Selects an interface, controller, and configures it as a network clock source at a particular priority. • **system**—Required for platforms that have an internal clock generator. Not applicable for the Cisco 7600 series routers. • *priority*—Configures the priority of network clock source. Values range from 1 to 6. • *interface name*—Configures the network-clock-source to the selected interface. • **global**—Configures the network clock to use a global configuration. • **local**—Configures the network clock to use a local configuration. **Note** Configure only one source at a time.
Router# **[no] network-clock participate** *slotnum*	Enables a line card to participate in network clocking feature. This is default mode. The **no** form of this command prevents a line card from participating in network clocking feature. When a slot is disabled, it can neither source nor take the clock from the backplane.

Command or Action	Purpose
Router# **[no] network-clock revertive**	Configures revertive behavior on the network clock. When revertive mode is configured and a previously unavailable higher priority source comes up, then this source becomes the active clock and the previous active source becomes the standby clock. Revertive mode is the default mode and is applicable for all types of interface failures. The alternate source is selected only if there is an interface failure, the alternate source is not selected when a source is supplying the bad clock. The **no** form of this command configures nonrevertive mode.
Router(config)# [no] **network-clock switchover marginal-source**	Prevents an interface from sending an OOR clock. A clock that exceeds the +/-9.2 ppm threshold goes into an OOR state and next alternate source is selected as active. Use the **no** form of this command to disable it. The default is that switchover occurs on a bad clock.
Router# **clock source** {**line** | **internal** | **network**}	Enables network clocking and configures clocking on the interface. • **line—**Specifies clock recovered from line • **internal—**Specifies SPA internal clock or clock from the host • **network**—Specifies network clock or the host card's internal oscillator
Router# **show network-clocks**	Displays details about the configured clocks and the current operational clocks and provides status information.
Router# **show platform hardware network-clocks**	Shows the mode of operation of the line cards along with relevant SONET clock register settings. This command is available for line card consoles only.
Router# **debug network-clock**	This command when enabled helps in debugging network clocking feature operation.
Router# **debug network-clock redundancy**	Enables high availability (HA) related debugging.

Verifying

Use the **show platform hardware network-clocks** command to verify.

```
SIP-400-4# show plat hardware network-clocks
SONET Clock Register = 0x20CA8000
SONET Clock Interrupt Enable Register = 0x0
SONET Clock Interrupt Status Register = 0x0

MT90401 Reference : Primary  Free Running
        Primary   : SPA 0
        Secondary : SPA 0
Backplane Reference
        Primary   DISABLED  : SPA 0
        Secondary DISABLED  : SPA 0
Status :
        Lock : 0  HoldOver : 0  SecOOR : 1 PriOOR : 1
        CLK_2M_OK : 1
```

```
Config :
        PCCI : 0  FLOCK : 0  ModeSel : 2
        SI5321 CAL Signal : 0 SI5321 LOS Signal : 0
        SI5321 HoldOver : 0
SIP-400-4#

use the show network-clock command to verify output on RP

Router# show network-clocks
 Active source = SONET 1/3/0
 Active source backplane reference line = Primary Backplane Clock


 All Network Clock Configuration
---------------------------------
 Priority    Clock Source             State                            Reason
 1           SONET 1/3/0              Valid

 Current operating mode is Revertive

 Current OOR Switchover mode is Switchover

There are no slots disabled from participating in network clocking
```

Configuring Clock Recovery

When configuring clock recovery, consider the following guidelines:

Adaptive Clock Recovery

- Clock source:
  - In Cisco IOS Release 12.2(33)SRC and later, both the 1-Port Channelized OC-3/STM-1 ATM CEoP SPA and the 24-Port Channelized T1/E1 ATM CEoP SPA can be used as a clock source.
  - In earlier releases, only the 24-Port Channelized T1/E1 ATM CEoP SPA can be a clock source.
- Number of clock sources allowed:
  - In Cisco IOS Release 12.2(33)SRC and later, multiple clocks can be sourced for the router: one clock for each SPA.
  - In earlier releases, only a single clock can be sourced for a router.
- The clock must be the same as used by the router as the network clock. Any pseudowire in this case can carry the clock.
- The minimum bundle size of CEM pseudowires on the network that delivers robust clock recovery is 4 DS0s.
- The minimum packet size of CEM pseudowires on the network that delivers robust clock recovery is 64 bytes.

Differential Clocking

- The maximum number of differential clocks sourced from a 24-Port Channelized T1/E1 ATM CEoP SPA is 24.
- The 24-Port Channelized T1/E1 ATM CEoP SPA can recover up to 24 T1/E1 clocks.
- There are several bundles sent from the same port. The bundle that is used for carryingthe clock of the port is the first created bundle of the port. Only pseudowires that include the first DS0 of a port can carry differential clock.

To configure clock recovery on a 24-Port Channelized T1/E1 ATM CEoP SPA, use the following procedure:

	Command or Action	Purpose
Step 1	Router(config)# **controller** {**e1** | **t1**} *slot*/*subslot*/*port*	Selects the controller.
Step 2	Router(config-controller)# **recovered-clock** *slot*/*subslot*	Specifies the interface for the recovered clock.
Step 3	Router(config-controller)# **clock recovered** *clock-id* {**adaptive** | **differential**} **cem** *port cem-group*	Specifies the recovered clock number and the clock recovery type.
Step 4	Router(config-controller)# **clock reference** {**enhanced** | **internal**}	Specifies the clock reference.
Step 5	Router(config-controller)# **clock master**	Configures the clock master.
Step 6	Router(config-controller)# **clock slave**	Configures the clock slave.

To apply the recovered clock to the controller, use the following procedure:

	Command or Action	Purpose
Step 1	Router(config)# **controller** {**e1** | **t1**} *slot*/*subslot*/*port*	Selects the controller.
Step 2	Router(config-controller)# **clock source recovered** *number*	Assigns a number to the recovered clock.
Step 3	Router(config-controller)# **cem-group** *number* **timeslots** *number*	Creates a circuit emulation channel from one or more time slots of a T1 or E1.
Step 4	Router(config-controller)# **recovered-clock** *slot*/*subslot*	Applies the recovered clock to the interface.
Step 5	Router(config-controller)# **clock recovered** *clock-id* {**adaptive** | **differential**} **cem** *port cem-group*	Specifies the recovered clock number and the clock recovery type.

Verifying Clock Recovery

To verify clock recovery, use the **show recovered-clock** command. In Cisco IOS Release 12.2SRB1 and later, command output has been expanded to include the port number and CEM group number.

```
Router# show recovered-clock
Recovered clock status for subslot 3/0
----------------------------------------
Clock    Mode         Port CEM  Status     Frequency Offset(ppb)
1        ADAPTIVE     0    1    HOLDOVER   0

Router# show recovered-clock
Recovered clock status for subslot 3/0
----------------------------------------
Clock    Mode         Port CEM  Status     Frequency Offset(ppb)
1        ADAPTIVE     0    1    ACQUIRING  -694
```

Use the **show platform network-clock** command to display the contents of network clocking registers.

```
Router# show platform network-clock
SONET Clock Register = 0x20EB80C8
SONET Clock Interrupt Enable Register = 0x0
SONET Clock Interrupt Status Register = 0x2
```

```
MT90401 Reference : Primary  Reserved
        Primary   : SPA 0
        Secondary : SPA 0
Backplane Reference
        Primary   ENABLE  : SPA 0
        Secondary ENABLE  : MT90401
Status :
        Lock : 0  HoldOver : 1  SecOOR : 1 PriOOR : 1
        CLK_2M_OK : 1
Config :
        PCCI : 0  FLOCK : 0  ModeSel : 3
        SI5321 CAL Signal : 0 SI5321 LOS Signal : 0
        SI5321 HoldOver : 0
```

Configuring Out-of-Band Clocking

A TDM network requires a synchronized clock at each end of the connection (the source and destination). This means that the source and destination clock signals must be synchronized to each other in order to maintain data integrity on the communication link.

On the other hand, a packet-switched network (PSN) does not use a clocking strategy, which means that the PSN does not provide frequency synchronization between source and destination routers. Therefore, to transmit TDM data across a PSN (such as an MPLS network), we need a way to deliver the clocking signal between the source and destination routers.

Out-of-band clocking provides a way to deliver a clock signal between two CEoP SPAs, which allows TDM devices connected to the SPAs to communicate with each other. Dedicated pseudowires (called out-of-band clock channels) carry the timing signal between the sending and receiving SPAs. When a TDM device sends data to a destination TDM device, the receiving SPA uses the out-of-band clock channel to recover the clock signal that was used to send the data.

By keeping the timing packets separate from data packets, out-of-band clocking delivers an extremely accurate timing signal. This timing accuracy is important for mobile wireless applications and other specialized applications that have very low tolerances for such things as packet delay variation (PDV), jitter, and latency in the network. In-band clocking (where timing information is derived from the data stream) does not provide a clock that is accurate enough for these applications.

To set up out-of-band clock channels, you must configure a master clock interface and a slave clock interface on the SPAs and configure pseudowires to connect the master and slave clocks. Instructions for performing these steps are provided later in this section.

Benefits

Out-of-band clocking provides the following benefits:

- Enables mobile wireless providers to migrate from TDM networks to PSNs in order to save on costs and improve scalability.
- CEoP equipment can ignore the contents of the timing packets that are sent over the out-of-band clock channel because the packets do not contain data.
- Allows the CEoP SPA to be used for applications that use something other than constant bit rate (CBR) data. For example, out-of-band clocking allows the SPA to be used for 3G (data) wireless applications, which use AAL2 in variable bit rate (VBR) mode. In addition, out-of-band clocking allows the SPA to be used for 2G (voice) applications.

- Provides recovered clock accuracy that complies with ITU-T specifications G.823 and G.824, which enables the CEoP SPA to be used in mobile and wireless applications (including voice) that require extreme synchronization accuracy.
- Provides an alternative clock-recovery mechanism when adaptive clocking cannot be deployed.
- Enables the CEoP SPA to be the master clock in a PSN.
- Makes it possible to have two master clocks. Previously, only one master clock was possible.

Configuration Guidelines

The following guidelines apply to out-of-band clocking on CEoP SPAs:

- The default packet size for out-of-band clock channels (CEM circuits) is 910 bytes.
- Out-of-band clocking can co-exist with Stateful SwitchOver (SSO), but it is not SSO compliant. Therefore, if a switchover occurs, the out-of-band clocking functionality is not available for a brief period of time while the feature is brought back online.
- A CEoP SPA cannot be configured as both a master and slave clock. To reconfigure a SPA's clock type, you must first remove the existing clock configuration (master or slave).
- Pseudowires for out-of-band clocking are configured under the virtual CEM interface that represents the recovered clock interface. This differs from normal CEM pseudowires, which are configured under the port (controller interface).
- The master clock pseudowire and slave clock pseudowire should be on different CEoP SPAs.

Router Sending Clock (Master Clock)

- You must select the common telecom 19.44MHz clock as the recovered clock to use for the master clock.
- A maximum of 64 out-of-band clock channels can be configured from the CEoP SPA that provides the master clock signal.
- The out-of-band clock channel (pseudowire) is configured under the virtual CEM interface that represents the SPA from which the master clock is recovered. The **xconnect** command used to create the clock channel must specify the destination for the clock signal.
- The out-of-band clock stream is sent in SAToP (unframed) format.

Router Recovering Clock (Slave Clock)

- The out-of-band clock signal is always recovered in adaptive mode. The clock signal can then be used to drive all of the ports on the CEoP SPA.
- Two CEM circuits (a primary and a secondary out-of-band channel) can be configured under a slave clock interface, one for each of two master clock signals. This way, the SPA can receive a master clock signal from two separate sources (that is, two master clocks).
- Under the slave clock interface, the **xconnect** command (used to create the out-of-band clock channel) must specify the router from which the master clock is recovered.

Configuration Overview

The following steps provide a high-level overview of the procedure for configuring out-of-band clocking between two CEoP SPAs. Detailed steps are provided in the sections that follow.

Before you begin, determine which CEoP SPAs have TDM devices connected to them. You must configure an out-of-band clock channel to deliver the clock signal from each SPA that sends TDM data to every destination SPA that receives the data.

1. Use the **recovered clock** command to identify the CEoP SPA that is to send TDM data across the MPLS network. This SPA's clock is used as the master clock for out-of-band clocking.
2. Configure master and slave clock interfaces to represent the source (clock master) and destination (clock slave) for the out-of-band clock signal. The master and slave clock interfaces (and pseudowires) should be configured on different SPAs.
   a. The master clock interface represents the master clock, which is distributed to all destination CEoP SPAs that receive data from the source TDM device connected to this SPA. (See the "Creating and Configuring the Master Clock Interface" section on page 10-36 for instructions.)
   b. Configure a slave clock interface on each of the SPAs connected to TDM devices that can receive data from the source TDM device. (See the "Configuring the Slave Clock Interface" section on page 10-37 for detailed instructions.)

Note When you configure a master or slave clock interface, the router creates a virtual CEM interface to represent this out-of-band clock. The virtual CEM interface has the same slot and subslot information as the CEoP SPA from which the master clock is recovered. The port number is always 24. For example, if the clock signal is recovered from the SPA in slot 8, subslot 1 (**recovered-clock 8 1**), the virtual CEM interface is virtual-cem8/1/24.

3. Under both the master and slave clock interfaces, use the **cem** *circuit-id* command to configure CEM circuits to represent the out-of-band channels that will distribute the clock signal over the MPLS network. Each CEM circuit represents a separate out-of-band channel for delivering the clock signal from the source (master clock) to a destination TDM device (slave clock). The out-of-band clock channel is created when you issue the **xconnect** command in the next step.
   – Under the master clock interface, you can configure up to 64 CEM circuits, one for each of the destination TDM devices that will use this clock signal as its master clock.
   – Under the slave clock interface (on the destination TDM device), you can configure one or two CEM circuits. Two CEM circuits are allowed because the clock slave can receive a clock signal from two master clocks.

   Note Each out-of-band clock channel requires two CEM circuits (one on the master clock interface and one on the slave clock interface). Each CEM circuit represents the CEM attachment circuit at one end of the out-of-band clock channel.

4. Create the out-of-band channel for the clock signal by using the **xconnect** command to configure two pseudowires between the CEM circuit on the master clock interface and the CEM circuit on the slave clock interface. The master clock pseudowire and slave clock pseudowire should be on different SPAs; however, you should use the same VCID for both pseudowires.
   a. Under the master clock interface, configure a pseudowire to the destination device (slave clock).
   b. Under the slave clock interface (on the SPA that connects to the destination TDM device), configure a pseudowire to the router that contains the master clock interface.

Creating and Configuring the Master Clock Interface

To create the master clock interface for out-of-band clocking, perform the following steps:

	Command or Action	Purpose
Step 1	Router(config)# **recovered-clock** *slot/subslot*	Specifies the slot and subslot of the CEoP SPA to recover the master clock signal from. This is the SPA from which the TDM data will be sent. **Note** You must specify the 19.44MHz clock as the recovered clock to use as the clock master.
Step 2	Router(config)# **clock master**	Specifies that the recovered clock is to be used as the master clock signal for out-of-band clocking. The router creates a virtual CEM interface for the master clock. Go to the following steps to configure an out-of-band channel to use for the master clock.

To configure the out-of-band channel to use for the master clock signal, perform the following steps:

	Command or Action	Purpose
Step 1	Router(config)# **int virtual-cem** *slot/subslot/port*	Selects the virtual CEM interface for the master clock and enters interface configuration mode. The interface has the same slot and subslot as the SPA from which the master clock was recovered (Step 1 in the preceding task), and the port number is always 24.
Step 2	Router(config-if)# **cem** *circuit-id*	Creates a CEM attachment circuit for the master clock signal. Valid values for *circuit-id* are 0 to 63. **Note** You can configure up to 64 CEM circuits under the master clock interface.
Step 3	Router(config-if-cem)# **xconnect** *peer-router-id vcid* **encapsulation mpls**	Configures an out-of-band channel (pseudowire) to carry the master clock signal. • *peer-router-id* is the IP address of the router that is connected to the destination TDM device. • *vcid* is a 32-bit identifier for the pseudowire. • **encapsulation mpls** sets MPLS for the tunneling mode. **Note** Use the same *vcid* for the master and slave clock pseudowires; otherwise, the clock channel does not come up.
Step 4	Router(config-if-cem-xconn)# **end**	Exits CEM interface configuration mode and returns you to privileged EXEC mode.

Note A CEoP SPA cannot be configured as both master and slave at the same time. To reconfigure a SPA's clock type, you must first remove the existing clock configuration.

Configuring the Slave Clock Interface

To configure the slave clock interface and out-of-band channel to use for out-of-band clocking, perform the following steps. Configure a slave clock interface on every CEoP SPA that receives TDM data from the SPA configured as the master clock in the preceding section.

	Command or Action	Purpose
Step 1	Router(config)# **recovered-clock** *slot/subslot*	Specifies the slot and subslot of the CEoP SPA from which the master clock is recovered.
Step 2	Router(config)# **clock slave**	Creates a virtual CEM interface to represent the clock slave for out-of-band clocking.
Step 3	Router(config)# **int virtual-cem** *slot/subslot/port*	Enters configuration mode for the virtual CEM interface that represents the clock slave. • *slot/subslot* is the slot and subslot of the SPA from which the master clock was recovered (Step 1 above). • *port* is always 24.
Step 4	Router(config-if)# **cem** *circuit-id*	Creates a CEM attachment circuit for the clock slave. The *circuit-id* value can be: • 0—The primary clock source. • 1—The secondary clock source. **Note** You can configure up to two CEM circuits, one for each of two master clock signals.
Step 5	Router(config-if-cem)# **xconnect** *peer-router-id vcid* **encapsulation mpls**	Configures an out-of-band channel (pseudowire) to carry the clock signal. • *peer-router-id* is the IP address of the router that is connected to the source TDM device. • *vcid* is a 32-bit identifier for the pseudowire. • **encapsulation mpls** sets MPLS for the tunneling mode. **Note** Use the same VCID for the master and slave clock pseudowires; otherwise, the clock channel does not come up.
Step 6	Router(config-if-cem-xconn)# **end**	Exits CEM interface configuration mode and returns you to privileged EXEC mode.

Verifying Out-of-Band Clocking

This section lists the **show** commands that you can use to verify the out-of-band clocking configuration.

- Use the **show ip interface brief** command to display the virtual CEM interfaces that the router created to represent master and slave clock interfaces. The output in the following example shows only the virtual CEM interface. Information for all other interfaces is omitted from the display.

```
Router# show ip int brief
. . .
Virtual-cem8/1/24      unassigned      YES unset  up                    up
. . .
```

- Use the **show cem circuit** command to display a list of CEM circuits configured on the SPA. The command displays both normal and out-of-band clocking CEM circuits.

```
Router# show cem circuit
CEM Int.       ID   Line      Admin     Circuit         AC
--------------------------------------------------------------
CEM8/1/1       1    DOWN      DOWN      Active          --/--
Virtual-cem8/1/1    DOWN      UP        Active           UP
```

- Use the **show cem interface virtual-cem** *slot/subslot/port* command to display information about a particular virtual CEM interface:

```
Router# show cem interface virtual-cem 8/1/24
(Virtual-cem8/1/24) State: CONFIG COMPLETE
Virtual CEM Slave Clock Interface
Slot 8, Slot Unit 88, VC -1
Total cem circuits:    1
Cem circuits up   :    1
Cem circuits down :    0
```

- Use the **show run interface virtual-cem** *slot/subslot/port* command to dislay the current running configuration for the specified interface:

```
Router# show run int virtual-cem 8/1/24
Building configuration...

Current configuration : 117 bytes
!
interface Virtual-cem8/1/24
 no ip address
 cem 1
  rtp-present
  xconnect 20.0.0.1 300 encapsulation mpls
 !
end
```

- Use the **show run | begin recovered** command to display the recovered clock being used for out-of-band clocking:

```
Router# show run | begin recovered
recovered-clock 8 1
 clock master
```

- On the clock slave, you can use the **show recovered-clock** command to display the status of the out-of-band clock:

```
Router# show recovered-clock
Recovered clock status for subslot 3/0
----------------------------------------
Clock    Mode          Port CEM  Status     Frequency Offset(ppb)
ENHANCED PRIMARY           0    HOLDOVER   0
```

Removing the Out-of-Band Clocking Configuration

Use the following commands to delete the various components used for out-of-band clocking:

- To remove a CEM circuit, use the **no cem** *circuit-id* command (where *circuit-id* is the number assigned to the circuit). Issue the command under the virtual CEM interface where the circuit exists.

```
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# int virtual-cem 8/1/24
Router(config-if)# no cem 1
Router(config-if)# end
```

- To remove a virtual CEM interface, use the **no clock master** or **no clock slave** command in recovered-clock configuration mode, as shown in the following examples. Note that the virtual CEM interface is not deleted when you remove the last CEM circuit under the interface.

```
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# recovered-clock 8 1
Router(config-clock)# no clock master
Router(config-clock)# end
Router#
```

  In the following example, the **no clock slave** command deletes the slave clock interface for the recovered clock (which is 8/1):

```
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# recovered-clock 8 1
Router(config-clock)# no clock slave
Router(config-clock)# end
Router#
```

Out-of-Band Clocking Configuration Example

This section provides an example of how to configure out-of-band clocking between two CEoP SPAs. It is divided into several different configuration sections.

Configuring the Master Clock Interface

The following example shows how to configure a CEoP SPA as a master clock and verify the configuration:

```
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router (config)# recovered-clock ?
  <0-14>  Slot number

Router (config)# recovered-clock 8 1
Router(config-clock)# clock ?
  master     Configure clock master on the card
  recovered  Configure recovered clock on the card
  reference  Configure reference clock on the card
  slave      Configure clock slave on the card
Router(config-clock)# clock master
Router(config-clock)# end

Router# show run | begin recovered
recovered-clock 8 1
 clock master
```

Configuring the Slave Clock Interface

```
Router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# recovered-clock 8 1
Router(config-clock)# clock slave
Router(config-clock)# end
Router#
Router# show run | begin recovered-clock
recovered-clock 8 1
 clock slave
```

Verifying the Virtual CEM Interface Configuration

The router creates a virtual CEM interface when you configure either the master or slave clock interface. You can view the interface using the **show ip interface brief** command:

```
Router# show ip int br
…
Virtual-cem8/1/24      unassigned      YES unset  up                    up
…

Router# sh run int Virtual-cem 8/1/24
Building configuration...

Current configuration : 50 bytes
!
interface Virtual-cem8/1/24
 no ip address
end
```

Configuring CEM Circuits for Out-of-Band Clocking Example

This section provides an example of how to configure CEM circuits and pseudowires for out-of-band clocking. The sample configuration shows the circuits and pseudowires configured on a CEoP SPA in PE1, which sends TDM data to another CEoP SPA in PE2.

You configure CEM circuits for the master and slave clocks under the virtual CEM interface that represents the recovered clock that is being used for out-of-band clocking. This differs from normal CEM circuits, which are configured under the SPA controller through the cem-group command.

Issuing the **xconnect** command under the master and slave CEM circuits configures an out-of-band clock channel to use to send the clock signal from the sending SPA to the receiving SPA. Note that normal CEM pseudowires are configured under the SPA controller interface.

Out-of-Band Clocking (PE1)

```
PE1# conf t
PE1(config)# int virtual-cem 8/1/24
PE1(config-if)# cem 1
PE1(config-if-cem)# xconnect 20.0.0.1 200 encap mpls
PE1(cfg-if-cem-xconn)# end

PE1# show run int Virtual-CEM 8/1/24
Building configuration...

Current configuration : 117 bytes
!
interface Virtual-cem8/1/24
 no ip address
 cem 1
  rtp-present
  xconnect 20.0.0.1 200 encapsulation mpls
```

```
 !
end
```

Out-of-Band Clocking (PE2)

```
PE2# conf t
PE2(config)# int virtual-cem 8/1/24
PE2(config-if)# cem 1
PE2(config-if-cem)# xconnect 10.0.0.1 200 encap mpls
PE2(cfg-if-cem-xconn)# end

PE2# show run int Virtual-CEM 8/1/24
Building configuration...

Current configuration : 117 bytes
!
interface Virtual-cem8/1/24
 no ip address
 cem 1
  rtp-present
  xconnect 10.0.0.1 200 encapsulation mpls
 !
end
```

Configuring CEM Parameters

The following sections describe the parameters you can configure for CEM circuits.

Note The CEM parameters at the local and remote ends of a CEM circuit must match; otherwise, the pseudowire between the local and remote PE routers will not come up.

Configuring Payload Size (Optional)

To specify the number of bytes encapsulated into a single IP packet, use the **pay-load** *size* command. The *size* argument specifies the number of bytes in the payload of each packet. The range is from 32 to 1313 bytes.

Default payload sizes for an unstructured CEM channel are as follows:

- E1 = 56 bytes
- T1 = 192 bytes
- T3/E3 = 1024 bytes

Default payload sizes for a structured CEM channel depend on the number of time slots that constitute the channel. Payload size (L in bytes), number of time slots (N), and packetization delay (D in milliseconds) have the following relationship: L = 8*N*D. The default payload size is selected in such a way that the packetization delay is always 1 millisecond. For example, a structured CEM channel of 16xDS0 has a default payload size of 128 bytes.

The payload size must be an integer of the multiple of the number of time slots for structured CEM channels.

Setting the Dejitter Buffer Size

To specify the size of the dejitter buffer used to compensate for the network filter, use the **dejitter-buffer** *size* command. The configured dejitter buffer size is converted from milliseconds to packets and rounded up to the next integral number of packets. Use the *size* argument to specify the size of the buffer, in milliseconds. The range is from 1 to 500 ms; the default is 5 ms.

Setting the Idle Pattern (Optional)

To specify the idle pattern, use the [**no**] **idle-pattern** *pattern1* command. The payload of each lost CESoPSN data packet must be replaced with the equivalent amount of the replacement data. The range for *pattern* is from 0x0 to 0xFF; the default idle pattern is 0xFF.

Enabling Dummy Mode

Dummy mode enables a bit pattern for filling in for lost or corrupted frames. To enable dummy mode, use the **dummy-mode** [**last-frame** | **user-defined**] command. The default is last-frame. The following is an example:

```
Router(config-cem)# dummy-mode last-frame
```

Setting the Dummy Pattern

If dummy mode is set to user defined, you must use the **dummy-pattern** *pattern* command to configure the dummy pattern. The range for *pattern* is from 0x0 to 0xFF. The default dummy pattern is 0xFF. The following is an example:

```
Router(config-cem)# dummy-pattern 0x55
```

Shutting Down a CEM Channel

To shut down a CEM channel, use the **shutdown** command in CEM configuration mode. The **shutdown** command is supported only under CEM mode and not under the CEM class.

Verifying the Interface Configuration

The **show cem circuit** command shows information about the circuit state, administrative state, the CEM ID of the circuit, and the interface on which it is configured. If **xconnect** is configured under the circuit, the command output also includes information about the attached circuit.

```
Router# show cem circuit ?
  <0-504>    CEM ID
  detail     Detailed information of cem ckt(s)
  interface  CEM Interface
  summary    Display summary of CEM ckts
  |          Output modifiers
```

```
Router# show cem circuit
CEM Int.       ID   Line      Admin     Circuit         AC
--------------------------------------------------------------
CEM1/1/0       1    UP        UP        ACTIVE          --/--
CEM1/1/0       2    UP        UP        ACTIVE          --/--
CEM1/1/0       3    UP        UP        ACTIVE          --/--
CEM1/1/0       4    UP        UP        ACTIVE          --/--
CEM1/1/0       5    UP        UP        ACTIVE          --/--
```

The **show cem circuit** *0-504* command displays the detailed information about that particular circuit.

```
Router# show cem circuit 1
CEM1/1/0, ID: 1, Line State: UP, Admin State: UP, Ckt State: ACTIVE
Idle Pattern: 0xFF, Idle cas: 0x8, Dummy Pattern: 0xFF
Dejitter: 5, Payload Size: 40
Framing: Framed, (DS0 channels: 1-5)
Channel speed: 56
CEM Defects Set
Excessive Pkt Loss RatePacket Loss

Signalling: No CAS
RTP: No RTP
Ingress Pkts:    25929               Dropped:             0
Egress Pkts:     0                   Dropped:             0
CEM Counter Details
Input Errors:    0                   Output Errors:       0
Pkts Missing:    25927               Pkts Reordered:      0
Misorder Drops:  0                   JitterBuf Underrun:  1
Error Sec:       26                  Severly Errored Sec: 26
Unavailable Sec: 5                   Failure Counts:      1
Pkts Malformed:  0
```

The **show cem circuit summary** command displays the number of circuits which are up or down per interface basis.

```
Router# show cem circuit summary
CEM Int.       Total Active  Inactive
--------------------------------------
CEM1/1/0       5     5       0
```

The **show running module** command shows detail on each CEM group:

```
Router# show running module 1
Building configuration...

Current configuration : 1542 bytes
card type t1 1 1
!
Controller T1 1/1/0
 framing esf
 linecode b8zs
 cem-group 1 timeslots 1-5 speed 56
 cem-group 2 timeslots 6-10 speed 56
 cem-group 3 timeslots 11-15 speed 56
 cem-group 4 timeslots 16-20 speed 56
 cem-group 5 timeslots 21-24 speed 56
!
Controller T1 1/1/1
 framing esf
 linecode b8zs
!
```

```
Controller T1 1/1/2
 framing esf
 linecode b8zs
!
Controller T1 1/1/3
 framing esf
!
Controller T1 1/1/4
 framing esf
 linecode b8zs
!
Controller T1 1/1/5
 framing esf
 fdl both
 linecode b8zs
!
Controller T1 1/1/6
 framing esf
 linecode b8zs
!
Controller T1 1/1/7
 framing esf
 linecode b8zs
!
Controller T1 1/1/8
 framing esf
 linecode b8zs
!
Controller T1 1/1/9
 framing esf
 clock source internal
 linecode b8zs
!
Controller T1 1/1/10
 framing esf
 linecode b8zs
!
Controller T1 1/1/11
 framing esf
 linecode b8zs
!
Controller T1 1/1/12
 framing esf
 linecode b8zs
!
Controller T1 1/1/13
 framing esf
 linecode b8zs
!
Controller T1 1/1/14
 framing esf
 linecode b8zs
!
Controller T1 1/1/15
 framing esf
 linecode b8zs
!
Controller T1 1/1/16
 framing esf
 linecode b8zs
!
Controller T1 1/1/17
 framing esf
 linecode b8zs
```

```
!
Controller T1 1/1/18
 framing esf
 linecode b8zs
!
Controller T1 1/1/19
 framing esf
 linecode b8zs
!
Controller T1 1/1/20
 framing esf
 linecode b8zs
!
Controller T1 1/1/21
 framing esf
 linecode b8zs
!
Controller T1 1/1/22
 framing esf
 linecode b8zs
!
Controller T1 1/1/23
 framing esf
 linecode b8zs
!
interface CEM1/1/0
 no ip address
 cem 1
 !
 cem 2
 !
 cem 3
 !
 cem 4
 !
 cem 5
 !
end

Router# show int cem 2/1/3
CEM2/1/3 is up, line protocol is up
  Hardware is Circuit Emulation Interface
  MTU 1500 bytes, BW 10000000 Kbit, DLY 0 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation CEM, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out

Router# show class cem class1
Class: class1
Idle Pattern: 0x9, Idle cas: 0xF
```

```
Dejitter: 5, Payload Size: 100
RTP: No RTP

Router# show class cem all
Class: abcdefghijklmn
Idle Pattern: 0xF, Idle cas: 0x8
Dejitter: 200, Payload Size: 200
RTP: Configured, RTP-HDR Compression: Disabled

Class: class1
Idle Pattern: 0x9, Idle cas: 0xF
Dejitter: 5, Payload Size: 100
RTP: No RTP

Class: 1234
Idle Pattern: 0xF, Idle cas: 0x8
Dejitter: 5, Payload Size: 32
RTP: No RTP
Router# show class cem detail
Class: abcdefghijklmn
Idle Pattern: 0xF, Idle cas: 0x8
Dejitter: 200, Payload Size: 200
RTP: Configured, RTP-HDR Compression: Disabled

Circuits inheriting this Class:
 None

Interfaces inheriting this Class:
 None

Class: class1
Idle Pattern: 0x9, Idle cas: 0xF
Dejitter: 5, Payload Size: 100
RTP: No RTP

Circuits inheriting this Class:
 None

Interfaces inheriting this Class:
 None

Class: 1234
Idle Pattern: 0xF, Idle cas: 0x8
Dejitter: 5, Payload Size: 32
RTP: No RTP

Circuits inheriting this Class:
 None

Router# show class cem class1
Class: class1
Idle Pattern: 0x9, Idle cas: 0xF
Dejitter: 5, Payload Size: 100
RTP: No RTP
```

PART 5

Ethernet Shared Port Adapters

CHAPTER 11

Overview of the Ethernet SPAs

This chapter provides an overview of the release history, and feature and Management Information Base (MIB) support for the Fast Ethernet and Gigabit Ethernet SPAs on the Cisco 7600 series router.

This chapter includes the following sections:

Release History

Release	Modification
12.2(33)SRC	• Added SFP-GE-T Support • Added SPA-1X10GE-L-V2 support to the SIP-400
12.2(33)SRB1	The Any Transport over MPLS over GRE (AToMoGRE) feature was introduced on the Cisco 7600 SIP-400 on the Cisco 7600 series router. The Backup Interface for Flexible UNI feature was introduced on the Cisco 7600 SIP-400 for Gigabit Ethernet SPAs.
12.2(33)SRA	Support for the following SPAs was introduced on the Cisco 7600 SIP-200 on the Cisco 7600 series router: • 4-Port Fast Ethernet SPA • 8-Port Fast Ethernet SPA The Multipoint Bridging feature was introduced on the Cisco 7600 SIP-400 on the Cisco 7600 series router. The Scalable EoMPLS feature was increased from 4 K to 12 K on the Cisco 7600 SIP-400 on the Cisco 7600 series router. Support for Ethernet Connectivity Fault Management and Ethernet Operations, Administration, and Maintenance was introduced.

12.2(18)SXF	Support for the following SPAs was introduced on the Cisco 7600 SIP-600 on the Cisco 7600 series router and Catalyst 6500 series switch: • 1-Port 10-Gigabit Ethernet SPA • 5-Port Gigabit Ethernet SPA • 10-Port Gigabit Ethernet SPA Support for the following SPA was introduced on the Cisco 7600 SIP-400 on the Cisco 7600 series router and Catalyst 6500 series switch: • 2-Port Gigabit Ethernet SPA

Supported Features

The following is a list of some of the significant hardware and software features supported by the Fast Ethernet and Gigabit Ethernet SPAs on the Cisco 7600 series router:

- Autonegotiation
- Full-duplex operation
- 802.1Q VLAN termination
- Jumbo frames support (9216 bytes)
- Support for command-line interface (CLI)-controlled OIR
- 802.3x flow control
- Up to 4000 VLANs per SPA
- Up to 5000 MAC Accounting Entries per SPA (Source MAC Accounting on the ingress and Destination MAC Accounting on the egress)
- Per-port byte and packet counters for policy drops, oversubscription drops, CRC error drops, packet sizes, unicast, multicast, and broadcast packets
- Per-VLAN byte and packet counters for policy drops, oversubscription drops, unicast, multicast, and broadcast packets
- Per-port byte counters for good bytes and dropped bytes
- Multiprotocol Label Switching (MPLS)
- Any Transport over MPLS over GRE (AToMoGRE)
- Ethernet over Multiprotocol Label Switching (EoMPLS)
- Quality of service (QoS)
- Hot Standby Router Protocol (HSRP)
- Virtual Router Redundancy Protocol (VRRP)
- User-set speed
- Hierarchal Virtual Private LAN Service (H-VPLS) (Gigabit Ethernet SPAs only)
- Multipoint Bridging (Gigabit Ethernet SPAs only)
- Connectivity Fault Management (CFM)
- IP Subscriber Awareness over Ethernet

Restrictions

Note For other SIP-specific features and restrictions see also Chapter 3, "Overview of the SIPs and SSC."

The following restrictions apply to Cisco IOS Release 12.2(18)SXF:

- EtherChannel is not supported on Fast Ethernet SPAs or the 2-Port Gigabit Ethernet SPA on the Cisco 7600 SIP-400.

Supported MIBs

The following MIBs are supported by the Fast Ethernet and Gigabit Ethernet SPAs on the Cisco 7600 series router:

- ENTITY-MIB (RFC 2737)
- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- CISCO-ENTITY-ALARM-MIB
- CISCO-ENTITY-SENSOR-MIB
- IF-MIB
- ETHERLIKE-MIB (RFC 2665)
- Remote Monitoring (RMON)-MIB (RFC 1757)
- CISCO-CLASS-BASED-QOS-MIB
- MPLS-related MIBs
- Ethernet MIB/RMON

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

SPA Architecture

This section provides an overview of the architecture of the Fast Ethernet and Gigabit Ethernet SPAs and describes the path of a packet in the ingress and egress directions. Some of these areas of the architecture are referenced in the SPA software and can be helpful to understand when troubleshooting or interpreting some of the SPA CLI and **show** command output.

Every incoming and outgoing packet on the Fast Ethernet SPAs goes through the physical port (PHY RJ45), the Media Access Controller (MAC), and a Layer 2 Filtering/Accounting ASIC. Every incoming and outgoing packet on the Gigabit Ethernet SPAs goes through the physical (PHY) SFP optics, the Media Access Controller (MAC), and a Layer 2 Filtering/Accounting ASIC.

Path of a Packet in the Ingress Direction

The following steps describe the path of an ingress packet through the Fast Ethernet or Gigabit Ethernet SPAs:

1. For Fast Ethernet SPAs, each of the ports receives incoming frames from one of the RJ45 interface connectors. For Gigabit Ethernet SPAs, the SFP optics receive incoming frames on a per-port basis from one of the optical fiber interface connectors.
2. For Fast Ethernet SPAs, the PHY device processes the frame and sends it over a serial interface to the MAC device. For Gigabit Ethernet SPAs, the SFP PHY device processes the frame and sends it over a serial interface to the MAC device.
3. The MAC device receives the frame, strips the CRCs, and sends the packet via the SPI 4.2 bus to the ASIC.
4. The ASIC takes the packet from the MAC devices and classifies the Ethernet information. CAM lookups based on etype, port, VLAN, and source and destination address information determine whether the packet is dropped or forwarded to the SPA interface.

Path of a Packet in the Egress Direction

The following steps describe the path of an egress packet from the SIP through the Fast Ethernet and Gigabit Ethernet SPAs:

1. The packet is sent to the ASIC using the SPI 4.2 bus. The packets are received with Layer 2 and Layer 3 headers in addition to the packet data.
2. The ASIC uses port number, destination MAC address, destination address type, and VLAN ID to perform parallel CAM lookups. If the packet is forwarded, it is forwarded via the SPI 4.2 bus to the MAC device.
3. For Fast Ethernet SPAs, the MAC device forwards the packets to the PHY RJ45 interface, which transmits the packet. For Gigabit Ethernet SPAs, the MAC device forwards the packets to the PHY laser-optic interface, which transmits the packet.

Displaying the SPA Hardware Type

To verify the SPA hardware type that is installed in your Cisco 7600 series router, you can use the **show interfaces** command.

Table 11-1 shows the hardware description that appears in the **show** command output for each type of Fast Ethernet and Gigabit Ethernet SPA that is supported on the Cisco 7600 series router.

Table 11-1 SPA Hardware Descriptions in show Commands

SPA	Description in show interfaces Command
4-Port Fast Ethernet SPA	Hardware is FastEthernet SPA
8-Port Fast Ethernet SPA	Hardware is FastEthernet SPA
1-Port 10-Gigabit Ethernet SPA	Hardware is TenGigEther SPA
2-Port Gigabit Ethernet SPA	Hardware is GigEther SPA
5-Port Gigabit Ethernet SPA	Hardware is GigEther SPA
10-Port Gigabit Ethernet SPA	Hardware is GigEther SPA

Example of the show interfaces Command

The following example shows output from the **show interfaces fastethernet** command on a Cisco 7600 series router with a 4-Port Fast Ethernet SPA installed in slot 3:

```
Router# show interfaces fastethernet3/2/3
FastEthernet3/2/3 is up, line protocol is up
    Hardware is FastEthernet SPA, address is 000e.d623.e840 (bia 000e.d623.e840)
    Internet address is 33.1.0.2/16
    MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
        reliability 255/255, txload 59/255, rxload 83/255
    Encapsulation ARPA, loopback not set
    Keepalive not supported
    Full-duplex, 100Mb/s
    ARP type: ARPA, ARP Timeout 04:00:00
    Last input 00:00:11, output 00:00:08, output hang never
    Last clearing of "show interface" counters 3d00h
    Input queue: 0/75/626373350/0 (size/max/drops/flushes); Total output drops: 0
    Queueing strategy: fifo
    Output queue: 0/40 (size/max)
    5 minute input rate 32658000 bits/sec, 68032 packets/sec
    5 minute output rate 23333000 bits/sec, 48614 packets/sec
        17792456686 packets input, 1067548381456 bytes, 0 no buffer
        Received 0 broadcasts (0 IP multicasts)
        0 runts, 0 giants, 0 throttles
        0 input errors, 0 CRC, 0 frame, 130043940 overrun, 0 ignored
        0 watchdog
        0 input packets with dribble condition detected
        12719598014 packets output, 763177809958 bytes, 0 underruns
        0 output errors, 0 collisions, 0 interface resets
        0 babbles, 0 late collision, 0 deferred
        0 lost carrier, 0 no carrier
        0 output buffer failures, 0 output buffers swapped out
```

The following example shows output from the **show interfaces gigabitethernet** command on a Cisco 7600 series router with a 2-Port Gigabit Ethernet SPA installed in slot 2:

```
Router# show interfaces gigabitethernet 2/0/1
GigabitEthernet2/0/1 is down, line protocol is down
  Hardware is GigEther SPA, address is 000a.f330.2e40 (bia 000a.f330.2e40)
  Internet address is 2.2.2.1/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
```

```
  Encapsulation ARPA, loopback not set
  Full-duplex, 1000Mb/s, link type is force-up, media type is SX
  output flow-control is on, input flow-control is on
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 03:19:34, output 03:19:29, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     1703 packets input, 638959 bytes, 0 no buffer
     Received 23 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 1670 multicast, 0 pause input
     1715 packets output, 656528 bytes, 0 underruns
     0 output errors, 0 collisions, 4 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 PAUSE output
     0 output buffer failures, 0 output buffers swapped out
```

The following example shows output from the **show interfaces tengigabitethernet** command on a Cisco 7600 series router with a 1-Port 10-Gigabit Ethernet SPA installed in slot 7:

```
Router# show interfaces tengigabitethernet7/0/0
TenGigabitEthernet7/0/0 is up, line protocol is up (connected)
  Hardware is TenGigEther SPA, address is 0000.0c00.0102 (bia 000f.342f.c340)
  Internet address is 15.1.1.2/24
  MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
  Full-duplex, 10Gb/s
  input flow-control is on, output flow-control is on
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 00:00:10, output hang never
  Last clearing of "show interface" counters 20:24:30
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  L2 Switched: ucast: 0 pkt, 0 bytes - mcast: 0 pkt, 0 bytes
  L3 in Switched: ucast: 0 pkt, 0 bytes - mcast: 0 pkt, 0 bytes mcast
  L3 out Switched: ucast: 0 pkt, 0 bytes mcast: 0 pkt, 0 bytes
     237450882 packets input, 15340005588 bytes, 0 no buffer
     Received 25 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 0 multicast, 0 pause input
     0 input packets with dribble condition detected
     1676 packets output, 198290 bytes, 0 underruns
     0 output errors, 0 collisions, 4 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 PAUSE output
     0 output buffer failures, 0 output buffers swapped out
```

CHAPTER 12

Configuring the Fast Ethernet and Gigabit Ethernet SPAs

This chapter provides information about configuring the 4-Port Fast Ethernet SPA (shared port adapter), 8-Port Fast Ethernet SPA, 1-Port 10-Gigabit Ethernet SPA, 2-Port Gigabit Ethernet SPA, 5-Port Gigabit Ethernet SPA, and 10-Port Gigabit Ethernet SPA on the Cisco 7600 series router. It includes the following sections:

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and the *Cisco IOS Configuration Fundamentals Command Reference* publications that correspond to your Cisco IOS software release.

Configuration Tasks

This section describes how to configure the Fast Ethernet and Gigabit Ethernet SPAs and includes information about verifying the configuration.

This section includes the following topics:

Required Configuration Tasks

This section lists the required configuration steps to configure the Fast Ethernet and Gigabit Ethernet SPAs. The commands in the section are applicable for both Fast Ethernet and Gigabit Ethernet SPAs; however, the examples below are for configuring a Gigabit Ethernet SPA. If you are configuring a Fast Ethernet SPA, replace the **gigabitethernet** command with the **fastethernet** command.

Some of the required configuration commands implement default values that might be appropriate for your network. If the default value is correct for your network, then you do not need to configure the command. These commands are indicated by "(As Required)" in the Purpose column.

Note Cisco Discovery Protocol (CDP) is disabled by default on Cisco 7600 SIP-400 interfaces.

To configure the Fast Ethernet or Gigabit Ethernet SPAs, complete the following steps:

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **interface fastethernet** *slot/subslot/port*[**.***subinterface-number*] or Router(config)# **interface gigabitethernet** *slot/subslot/port*[**.***subinterface-number*] or Router(config)# **interface tengigabitethernet** *slot/subslot/port*[**.***subinterface-number*]	Specifies the Fast Ethernet, Gigabit Ethernet or Ten Gigabit Ethernet interface to configure, where: • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4. • **.***subinterface-number*—(Optional) Specifies a secondary interface (subinterface) number.

	Command	Purpose
Step 3	Router(config-if)# **ip address** [*ip-address mask* {**secondary**} | **dhcp** {**client-id** *interface-name*}{**hostname** *host-name*}]	Sets a primary or secondary IP address for an interface that is using IPv4, where: • *ip-address*—Specifies the IP address for the interface. • *mask*—Specifies the mask for the associated IP subnet. • **secondary**—(Optional) Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address. • **dhcp**—Specifies that IP addresses will be assigned dynamically using DHCP. • **client-id** *interface-name*—Specifies the client identifier. The *interface-name* sets the client identifier to the hexadecimal MAC address of the named interface. • **hostname** *host-name*—Specifies the hostname for the DHCP purposes. The *host-name* is the name of the host to be placed in the DHCP option 12 field. **Note** The DHCP options with this command are not available for all Gigabit Ethernet SPAs and Fast Ethernet SPAs.
Step 4	Router(config-if)# **ip accounting mac-address** {**input** | **output**}	(Optional) Enables MAC address accounting. MAC address accounting provides accounting information for IP traffic based on the source and destination MAC addresses of the LAN interfaces, where: • **input**—Specifies MAC address accounting for traffic entering the interface. • **output**—Specifies MAC address accounting for traffic leaving the interface.
Step 5	Router(config-if)# **mtu** *bytes*	(As Required) Specifies the maximum packet size for an interface, where: • *bytes*—Specifies the maximum number of bytes for a packet. The default is 1500 bytes.

	Command	Purpose
Step 6	Router(config-if)# **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]	(Required for Hot Standby Router Protocol [HSRP] Configuration Only) Creates (or enables) the HSRP group using its number and virtual IP address, where: • *group-number*—(Optional) Specifies the group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • *ip-address*—(Optional on all but one interface if configuring HSRP) Specifies the virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • **secondary**—(Optional) Specifies the IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router. This command enables HSRP but does not configure it further. For additional information on configuring HSRP, refer to the HSRP section of the *Cisco IP Configuration Guide* publication that corresponds to your Cisco IOS software release.
Step 7	Router(config-if)# **no shutdown**	Enables the interface.

Specifying the Interface Address on a SPA

SPA interface ports begin numbering with “0” from left to right. Single-port SPAs use only the port number 0. To configure or monitor SPA interfaces, you need to specify the physical location of the SPA interface processor (SIP), SPA, and interface in the command-line-interface (CLI.) The interface address format is *slot*/*subslot*/*port*, where:

- *slot*—Specifies the chassis slot number in the Cisco 7600 series router where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA.

The following example shows how to specify the first interface (0) on a SPA installed in the first subslot of a SIP (0) installed in chassis slot 3:

```
Router(config)# interface serial 3/0/0
```

This command shows a serial SPA as a representative example, however the same *slot*/*subslot*/*port* format is similarly used for other SPAs (such as Asynchronous Transfer Mode [ATM] and packet over SONET [POS]) and other non-channelized SPAs.

Modifying the MAC Address on the Interface

The Gigabit Ethernet SPAs use a default MAC address for each port that is derived from the base address that is stored in the electrically erasable programmable read-only memory (EEPROM) on the backplane of the Cisco 7600 series router.

To modify the default MAC address of an interface to some user-defined address, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **mac-address** *ieee-address*	Modifies the default MAC address of an interface to some user-defined address, where: • *ieee-address*—Specifies the 48-bit IEEE MAC address written as a dotted triple of four-digit hexadecimal numbers (*xxxx.yyyy.zzzz*).

To return to the default MAC address on the interface, use the **no** form of the command.

Verifying the MAC Address

To verify the MAC address of an interface, use the **show interfaces gigabitethernet** privileged EXEC command and observe the value shown in the "address is" field.

The following example shows that the MAC address is 000a.f330.2e40 for interface 1 on the SPA installed in subslot 0 of the SIP installed in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces gigabitethernet 2/0/1
GigabitEthernet2/0/1 is up, line protocol is up
  Hardware is GigEther SPA, address is 000a.f330.2e40 (bia 000a.f330.2e40)
  Internet address is 2.2.2.1/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
  Full-duplex, 1000Mb/s, link type is force-up, media type is SX
  output flow-control is on, input flow-control is on
(Additional output removed for readability)
```

Gathering MAC Address Accounting Statistics

The **ip accounting mac-address** [**input** | **output**] command can be entered to enable MAC Address Accounting on an interface. After enabling MAC Address Accounting, MAC address statistics can be gathered by entering the **show interfaces mac-accounting** command.

Configuring HSRP

Hot Standby Router Protocol (HSRP) provides high network availability because it routes IP traffic from hosts without relying on the availability of any single router. HSRP is used in a group of routers for selecting an active router and a standby router. (An *active router* is the router of choice for routing packets; a *standby router* is a router that takes over the routing duties when an active router fails, or when preset conditions are met).

HSRP is enabled on an interface by entering the **standby** [*group-number*] **ip** [*ip-address* [**secondary**]] command. The **standby** command is also used to configure various HSRP elements. This document does not discuss more complex HSRP configurations. For additional information on configuring HSRP, see the refer to the HSRP section of the *Cisco IP Configuration Guide* publication that corresponds to your Cisco IOS software release.

In the following HSRP configuration, standby group 2 on GigabitEthernet port 2/1/0 is configured at a priority of 110 and is also configured to have a preemptive delay should a switchover to this port occur:

```
Router(config)# interface GigabitEthernet 2/1/0
Router(config-if)# standby 2 ip 120.12.1.200
Router(config-if)# standby 2 priority 110
Router(config-if)# standby 2 preempt
```

Verifying HSRP

To display HSRP information, use the **show standby** command in EXEC mode:

```
Router# show standby
Ethernet0 - Group 0
Local state is Active, priority 100, may preempt
Hellotime 3 holdtime 10
Next hello sent in 0:00:00
Hot standby IP address is 198.92.72.29 configured
Active router is local
Standby router is 198.92.72.21 expires in 0:00:07
Standby virtual mac address is 0000.0c07.ac00
Tracking interface states for 2 interfaces, 2 up:
UpSerial0
UpSerial1
```

Customizing VRRP

Customizing the behavior of Virtual Router Redundancy Protocol (VRRP) is optional. Be aware that as soon as you enable a VRRP group, that group is operating. It is possible that if you first enable a VRRP group before customizing VRRP, the router could take over control of the group and become the master virtual router before you have finished customizing the feature. Therefore, if you plan to customize VRRP, it is a good idea to do so before enabling VRRP.

To customize your VRRP configuration, use any of the following VRRP commands inTable 12-1 in interface configuration mode.

Table 12-1 VRRP Commands

Command	Purpose
Router(config-if)# **vrrp** *group* **authentication text** *text-string*	Authenticates VRRP packets received from other routers in the group. If you configure authentication, all routers within the VRRP group must use the same authentication string, where: • *group*—Virtual router group number for which authentication is being configured. The group number is configured with the **vrrp ip** command. • **text** *text-string*—Authentication string (up to eight alphanumeric characters) used to validate incoming VRRP packets.
Router(config-if)# **vrrp** *group* **description** *text*	Assigns a text description to the VRRP group, where: • *group*—Virtual router group number. • *text*—Text (up to 80 characters) that describes the purpose or use of the group.
Router(config-if)# **vrrp** *group* **priority** *level*	Sets the priority level of the router within a VRRP group. The default value is 100, where: • *group*—Virtual router group number. • *level* —Priority of the router within the VRRP group. The range is from 1 to 254. The default is 100.
Router(config-if)# **vrrp** *group* **preempt** [**delay** *seconds*]	Configures the router to take over as master virtual router for a VRRP group if it has a higher priority than the current master virtual router. This command is enabled by default. You can use it to change the delay, where: • *group*—Virtual router group number of the group for which preemption is being configured. The group number is configured with the **vrrp ip** command. • **delay** *seconds*—(Optional) Number of seconds that the router will delay before issuing an advertisement claiming master ownership. The default delay is 0 seconds.

Command	Purpose
Router(config-if)# **vrrp** *group* **timers advertise** [**msec**] *interval*	Configures the interval between successive advertisements by the master virtual router in a VRRP group, where: • *group*—Virtual router group number to which the command applies. • **msec**—(Optional) Changes the unit of the advertisement time from seconds to milliseconds. Without this keyword, the advertisement interval is in seconds. • *interval*—Time interval between successive advertisements by the master virtual router. The unit of the interval is in seconds, unless the **msec** keyword is specified. The default is 1 second.
Router(config-if)# **vrrp** *group* **timers learn**	Configures the router, when it is acting as backup virtual router for a VRRP group, to learn the advertisement interval used by the master virtual router, where: • *group*—Virtual router group number to which the command applies.

Enabling VRRP

To enable VRRP on an interface, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface** *type number*	Configures an interface, where: • *type*—Interface type. • *number*—Interface number.
Step 2	Router(config-if)# **vrrp** *group* **ip** *ipaddress*	Enables VRRP on an interface and identifies the primary IP address of the virtual router, where: • *group*—Virtual router group number to which the command applies. • *ipaddress*—IP address of the virtual router.
Step 3	Router(config-if)# **vrrp** *group* **ip** *ipaddress* **[secondary]**	(Optional) Enables VRRP on an interface. After you identify a primary IP address, you can use the **vrrp ip** command again with the **secondary** keyword to indicate additional IP addresses supported by this group, where: • *group*—Virtual router group number to which the command applies. • *ipaddress*—IP address of the virtual router. • **secondary**—(Optional) Indicates additional IP addresses supported by this group.

Verifying VRRP

To verify VRRP, use either of the following commands in EXEC mode:

Command	Purpose
Router# **show vrrp** [**brief** | *group*]	Displays a brief or detailed status of one or all VRRP groups on the router, where: • **brief**—(Optional) Provides a summary view of the group information. • *group*—(Optional) Virtual router group number of the group for which information is to be displayed. The group number is configured with the **vrrp ip** command.
Router# s**how vrrp interface** *type number* [**brief**]	Displays the VRRP groups and their status on a specified interface, where: • *type*—Interface type. • *number*—Interface number. • **brief**—(Optional) Provides a summary view of the group information.

Modifying the Interface MTU Size

The Cisco IOS software supports three different types of configurable maximum transmission unit (MTU) options at different levels of the protocol stack:

- Interface MTU—Checked by the SPA on traffic coming in from the network. Different interface types support different interface MTU sizes and defaults. The interface MTU defines the maximum packet size allowable (in bytes) for an interface before drops occur. If the frame is smaller than the interface MTU size, but is not smaller than the minimum frame size for the interface type (such as 64 bytes for Ethernet), then the frame continues to process.
- IP MTU—Can be configured on an interface or subinterface and is used by the Cisco IOS software to determine whether fragmentation of a packet takes place. If an IP packet exceeds the IP MTU size, then the packet is fragmented.
- Tag or Multiprotocol Label Switching (MPLS) MTU—Can be configured on an interface or subinterface and allows up to six different labels, or tag headers, to be attached to a packet. The maximum number of labels is dependent on your Cisco IOS software release.

Different encapsulation methods and the number of MPLS MTU labels add additional overhead to a packet. For example, Subnetwork Access Protocol (SNAP) encapsulation adds an 8-byte header, dot1q encapsulation adds a 2-byte header, and each MPLS label adds a 4-byte header (*n* labels x 4 bytes).

For the Fast Ethernet and Gigabit Ethernet SPAs on the Cisco 7600 series router, the default MTU size is 1500 bytes. When the interface is being used as a Layer 2 port, the maximum configurable MTU is 9216 bytes. The SPA automatically adds an additional 22 bytes to the configured MTU size to accommodate some of the additional overhead.

Interface MTU Configuration Guidelines

When configuring the interface MTU size on a Fast Ethernet and Gigabit Ethernet SPA on a Cisco 7600 series router, consider the following guidelines:

- The default interface MTU size accommodates a 1500-byte packet, plus 22 additional bytes to cover the following additional overhead:
  - Layer 2 header—14 bytes
  - Dot1q header—4 bytes
  - CRC—4 bytes

Note Depending on your Cisco IOS software release, a certain maximum number of MPLS labels are supported. If you need to support more than two MPLS labels, then you need to increase the default interface MTU size.

- If you are using MPLS, be sure that the **mpls mtu** command is configured for a value less than or equal to the interface MTU.
- If you are using MPLS labels, then you should increase the default interface MTU size to accommodate the number of MPLS labels. Each MPLS label adds 4 bytes of overhead to a packet.

Interface MTU Guidelines for Layer 2 Ports

On Layer 2 ports, it is important to understand the idea of the *jumbo MTU*. The *jumbo MTU* can be configured using the **system jumbomtu** command, although this command is only supported under the following scenarios:

- The port is a member of a Layer 2 EtherChannel.
- The new MTU size on the Layer 2 port is less than the currently configured maximum MTU for the port.

If neither of the above conditions applies to your configuration, neither does “jumbo MTU.”

Note Fast Ethernet SPAs cannot function as Layer 2 ports.

Interface MTU Configuration Task

To modify the MTU size on an interface, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **mtu** *bytes*	Configures the maximum packet size for an interface, where: • *bytes*—Specifies the maximum number of bytes for a packet. The default is 1500 bytes and the maximum configurable MTU is 9216 bytes.

To return to the default MTU size, use the **no** form of the command.

Verifying the MTU Size

To verify the MTU size for an interface, use the **show interfaces gigabitethernet** privileged EXEC command and observe the value shown in the MTU field.

The following example shows an MTU size of 1500 bytes for interface port 1 (the second port) on the Gigabit Ethernet SPA installed in the top subslot (0) of the SIP that is located in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces gigabitethernet 2/0/1
GigabitEthernet2/0/1 is up, line protocol is up
  Hardware is GigEther SPA, address is 000a.f330.2e40 (bia 000a.f330.2e40)
  Internet address is 2.2.2.1/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
```

Configuring the Encapsulation Type

By default, the interfaces on the Fast Ethernet and Gigabit Ethernet SPAs support Advanced Research Projects Agency (ARPA) encapsulation. They do not support configuration of service access point or SNAP encapsulation for transmission of frames; however, the interfaces will properly receive frames that use service access point and SNAP encapsulation.

The only other encapsulation supported by the SPA interfaces is IEEE 802.1Q encapsulation for virtual LANs (VLANs).

Configuring Autonegotiation on an Interface

Fast Ethernet and Gigabit Ethernet interfaces use a connection-setup algorithm called *autonegotiation.* Autonegotiation allows the local and remote devices to configure compatible settings for communication over the link. Using autonegotiation, each device advertises its transmission capabilities and then agrees upon the settings to be used for the link.

For the Fast Ethernet and Gigabit Ethernet interfaces on the Cisco 7600 series router, flow control is autonegotiated when autonegotiation is enabled. Autonegotiation is enabled by default.

The following guidelines should be followed regarding autonegotiation:

- If autonegotiation is disabled on one end of a link, it must be disabled on the other end of the link. If one end of a link has autonegotiation disabled while the other end of the link does not, the link will not come up properly on both ends.
- Autonegotiation is not supported on the 10-Port Gigabit Ethernet SPA on the Cisco 7600 SIP-600.
- Flow control can be configured separately of autonegotiation when Ethernet SPAs are installed in a Cisco 7600 SIP-600.
- Flow control is enabled by default.
- Flow control will be on if autonegotiation is disabled on both ends of the link.
- Flow control cannot be disabled on a Fast Ethernet SPA.

Disabling Autonegotiation

Autonegotiation is automatically enabled and can be disabled on the Fast Ethernet interfaces on the Cisco 7600 SIP-200, and the Gigabit Ethernet interfaces on the Cisco 7600 SIP-400 or Cisco 7600 SIP-600. During autonegotiation, advertisement for flow control, speed, and duplex occurs. If the Gigabit Ethernet interface is connected to a link that has autonegotiation disabled, autonegotiation should either be re-enabled on the other end of the link or disabled on the Fast Ethernet or Gigabit Ethernet SPA, if possible. Both ends of the link will not come up properly if only one end of the link has disabled autonegotiation.

Note Speed and duplex configurations are negotiated using autonegotiation. However, the only values that are negotiated are 100 Mbps for speed and full-duplex for duplex for Fast Ethernet SPAs, and 1000 Mbps for speed and full-duplex for duplex for Gigabit Ethernet SPAs. Therefore, from a user's perspective, these settings are not negotiated, but enabled using autonegotiation.

To disable autonegotiation on Fast Ethernet or Gigabit Ethernet SPAs, use the following commands in interface configuration mode:

Command	Purpose
Router(config-if)# **no negotiation auto**	Disables autonegotiation on a Fast Ethernet SPA interface on the Cisco 7600 SIP-200 or a Gigabit Ethernet SPA interfaces on the Cisco 7600 SIP-400. No advertisement of flow control occurs.
Router(config-if)# **speed nonegotiate**	Disables autonegotation of speed on Gigabit Ethernet SPA interfaces on the Cisco 7600 SIP-600.

Enabling Autonegotiation

Autonegotiation is automatically enabled and can be disabled unless it is on a SPA installed in a Cisco 7600 SIP-400, or on a 10-Port Gigabit Ethernet SPA, 5-Port Gigabit Ethernet SPA, or a 10-Port Gigabit Ethernet SPA when installed in a Cisco 7600 SIP-600. See the "Configuring Flow Control for an Ethernet SPA Interface on a Cisco 7600 SIP-600" section on page 12-22. To re-enable autonegotiation on a Fast Ethernet or Gigabit Ethernet interface, use the following commands in interface configuration mode:

Command	Purpose
Router(config-if)# **negotiation auto**	Enables autonegotiation on a Fast Ethernet SPA interface on a Cisco 7600 SIP-200 or a Gigabit Ethernet SPA interfaces on the Cisco 7600 SIP-400. Advertisement of flow control occurs.
Router(config-if)# **no speed nonegotiate**	Re-enables autonegotation on Gigabit Ethernet SPA interfaces on the Cisco 7600 SIP-600.

SFP-GE-T Support

The SFP-GE-T supports speeds of 10 Mbps, 100 Mbps, and 1000 Mbps. Speed is not autonegotiated; you must configure it using the **speed** command. Only full-duplex mode is supported.

Note Because autonegotiation of full-duplex is not supported, you must manually configure full-duplex mode.

You can configure each Ethernet interface independently using any combination of 10 Mbps, 100 Mbps, or 1000 Mbps.

To set the interface speed, use the following command in the interface configuration mode:

Command	Purpose
Router(config-if)# **speed {10 | 100 | 1000 | auto}**	Configures the interface speed. Accepted values are: • 10 for 10 Mbps operation • 100 for 100 Mbps operation • 1000 for 1000 Mbps operation

Configuring an Ethernet VLAN

For information on configuring Ethernet VLANs, see the "Creating or Modifying an Ethernet VLAN" section of the "Configuring VLANs" chapter in the *Cisco 7600 Series Cisco IOS Software Configuration Guide* publication that corresponds to your Cisco IOS software release.

Configuring a Subinterface on a VLAN

You can configure subinterfaces on the Fast Ethernet SPA interfaces and Gigabit Ethernet SPA interfaces on a VLAN using IEEE 802.1Q encapsulation. Cisco Discovery Protocol (CDP) is disabled by default on the 2-Port Gigabit Ethernet SPA interfaces and subinterfaces on the Cisco 7600 SIP-400.

To configure a SPA subinterface on a VLAN, use the following commands beginning in interface configuration mode:

Note On any Cisco 7600 SIP-600 Ethernet port subinterface using VLANs, a unique VLAN ID must be assigned. This VLAN ID cannot be in use by any other interface on the Cisco 7600 series router.

	Command	Purpose
Step 1	Router(config)# **interface fastethernet** *slot/subslot/port***.***subinterface-number* or Router(config)# **interface gigabitethernet** *slot/subslot/port***.***subinterface-number* or Router(config)# **interface tengigabitethernet** *slot/subslot/port***.***subinterface-number*	Specifies the Fast Ethernet, Gigabit Ethernet or Ten Gigabit Ethernet interface to configure, where: • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4. • **.***subinterface-number*—Specifies a secondary interface (subinterface) number.
Step 2	Router(config-subif)# **encapsulation dot1q** *vlan-id*	Defines the encapsulation format as IEEE 802.1Q ("dot1q"), where *vlan-id* is the number of the VLAN (1–4094).
Step 3	Router(config-if)# **ip address** *ip-address mask* [**secondary**]	Sets a primary or secondary IP address for an interface, where: • *ip-address*—Specifies the IP address for the interface. • *mask*—Specifies the mask for the associated IP subnet. • **secondary**—(Optional) Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.

Verifying Subinterface Configuration on a VLAN

To verify the configuration of a subinterface and its status on the VLAN, use the **show vlans** privileged EXEC command.

The following example shows the status of subinterface number 1 on port 0 on the SPA in VLAN number 200:

```
Router# show vlans
VLAN ID:200 (IEEE 802.1Q Encapsulation)

Protocols Configured:          Received:        Transmitted:
          IP                   0                   2

VLAN trunk interfaces for VLAN ID 200:

GigabitEthernet4/1/0.1 (200)

      IP:12.200.21.21

      Total 0 packets, 0 bytes input
      Total 2 packets, 120 bytes output
```

Configuring Layer 2 Switching Features

The Cisco 7600 series router supports simultaneous, parallel connections between Layer 2 Ethernet segments. After you review the SPA-specific guidelines described in this document, refer to the "Configuring Layer 2 Ethernet Interfaces" section of the *Cisco 7600 Series Router Cisco IOS Software Configuration Guide* for more information about configuring the Layer 2 switching features.

Configuring Multipoint Bridging

Multipoint bridging (MPB) enables the connection of multiple ATM PVCs, Frame Relay permanent virtual circuits (PVCs), Bridging Control Protocol (BCP) ports, and WAN Gigabit Ethernet subinterfaces into a single broadcast domain (virtual LAN), together with the LAN ports on that VLAN. This enables service providers to add support for Ethernet-based Layer 2 services to the proven technology of their existing ATM and Frame Relay legacy networks. Customers can then use their current VLAN-based networks over the ATM or Frame Relay cloud. This also allows service providers to gradually update their core networks to the latest Gigabit Ethernet optical technologies, while still supporting their existing customer base.

For MPB configuration guidelines and restrictions and feature compatibility tables, see the "Configuring Multipoint Bridging" section on page 4-23.

Configuring the Bridging Control Protocol

The Bridging Control Protocol (BCP) enables forwarding of Ethernet frames over SONET networks and provides a high-speed extension of enterprise LAN backbone traffic through a metropolitan area. The implementation of BCP on the SPAs includes support for IEEE 802.1D, IEEE 802.1Q Virtual LAN (VLAN), and high-speed switched LANs.

For BCP configuration guidelines and restrictions and feature compatibility tables, see the "Configuring PPP Bridging Control Protocol Support" section on page 4-35 of Chapter 4, "Configuring the SIPs and SSC."

Configuring AToM over GRE

MPLS over generic routing encapsulation (MPLSoGRE) encapsulates MPLS packets inside IP tunnels, creating a virtual point-to-point link across non-MPLS networks. This allows users of primarily MPLS networks to continue to use existing non-MPLS legacy networks until migration to MPLS is possible. AToM (any transport over MPLS) over GRE includes support the following transports:

- ATM over MPLS
- Frame Relay over MPLS (FRoMPLS)
- High-Level Data Link Control (HDLC) over MPLS
- Scalable Ethernet over MPLS (EoMPLS)
- Circuit Emulation over Packet (CEoP)
- Hardware-based EoMPLS

AToMoGRE is supported only on the following hardware:

- SIP-400, 5x1 GE SPA, 2x1 GE SPA (Core facing)
- ATM SPA (SPA-2xOC3-ATM, SPA-4xOC3-ATM, SPA-1xOC12-ATM, SPA-1xOC48-ATM, CEoPs SPA (such as OC3, 24T1/E1) with Inverse Multiplexing (IMA) support, and all Ethernet interfaces
- Sup32, Sup720, RSP720

AToMoGRE supports the following features:

- Provider edge (PE)-to-PE, P-to-PE, and P-to-P tunneling of MPLS packets (See Figure 12-1, Figure 12-2, and Figure 12-3.)

Figure 12-1 PE-to-PE GRE Tunnel

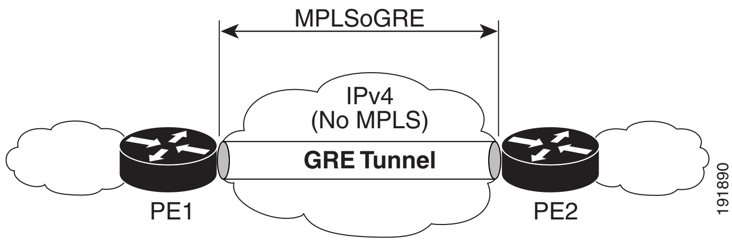


Figure 12-2 P-to-PE GRE Tunnel

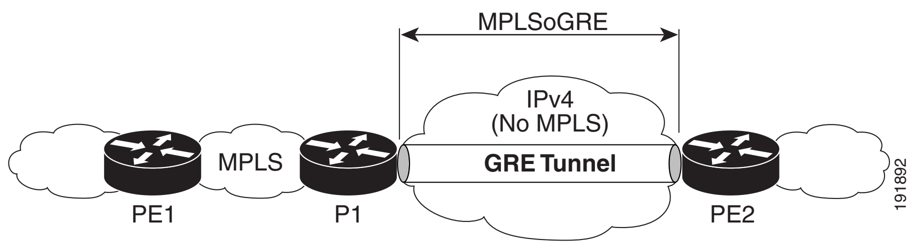
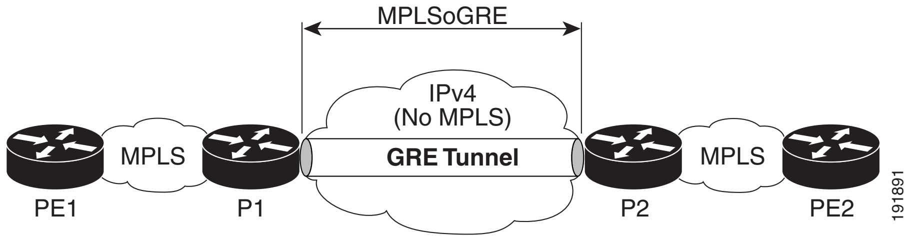


Figure 12-3 P-to-P GRE Tunnel



- IPv4 on customer edge (CE) facing interfaces.
- IPv4 on core facing interfaces.
- GRE 4-byte headers (no option fields).
- Nondedicated physical interface supporting both tunneled and nontunneled traffic.
- Multiple routes for the tunnel between the Cisco 7600 SIP-400 physical interface or subinterface and the IP cloud may exist. The routing protocol will pick only one route for MPLSoGRE traffic.
- No software-imposed limit on the maximum number of tunnels. The Cisco 7600 SIP-400 supports a maximum number of 128 tunnels. Tunnel traffic can be routed through Cisco 7600 SIP-400 main interfaces or subinterfaces.
- The Cisco 7600 SIP-400 physical interface or subinterface used for the tunnel endpoint can be used to carry native MPLS and AToMoMPLS and its variations: Hardware-based EoMPLS, FRoMPLS, PPPoMPLS, HDLCoMPLS, Scalable EoMPLS, and CEoP.

AToMoGRE Configuration Guidelines

The following guidelines apply to AToMoGRE:

- Ingress/egress features are not supported on the tunnel interface; they are supported on the physical interface or subinterface.
- Unsupported GRE options are: sequencing, checksum, key, source route.
- Some tunnel options are not supported: Carry Security Options of Client Packet, Unidirectional Link Routing, Mobile IP Path MTU Discovery.
- The Cisco 7600 SIP-400 physical interface or subinterface used for the tunnel endpoint cannot be used to carry Software-based EoMPLS and VPLS. Advanced features such as Carrier Supporting Carrier (CSC) and Inter-Autonomous Systems (Inter-AS) are not supported.
- AToM over GRE cannot be combined with the AToM Tunnel Select feature.

Configuring mVPNoGRE

The multicast Virtual Private Network over generic routing encapsulation (mVPNoGRE) provides a mechanism to send unicast and multicast packets across a non-MPLS network. This is accomplished by creating a GRE tunnel across the non-MPLS network. When MPLS (unicast VRF) or mVPN (multicast VRF) packets are sent across the non-MPLS network, they are encapsulated within a GRE packet and transverse the non-MPLS network through the GRE tunnel. Upon receiving the GRE packet at the other side of the non-MPLS network, it removes the GRE header and forwards the inner MPLS or unicast VRF or mVPN packet to its final destination.

Note For mVPNoGRE, there is one outer packet and two inner packets. The outer packet is unicast GRE. The first inner packet is multicast GRE (mVPN), and the second inner packet is normal (customer) multicast.

Note mVPNoGRE is not supported on Fast Ethernet SPAs on the Cisco 7600 SIP-200.

PE-to-PE Tunneling

mVPNoGRE uses the Provider Edge-to-Provider Edge (PE-to-PE) tunneling variation. mVPNoGRE provides a scalable way to connect multiple customer networks across a non-MPLS network. It does this by multiplexing traffic destined to multiple customer networks through a single GRE tunnel.

On each side of the non-MPLS network, each Customer Edge (CE) router is assigned a VPN Routing and Forwarding (VRF) number by the PE router. The IP networks behind the CE routers are learned by the PE router through a routing protocol such as BGP, OSPF or RIP. Routes to these networks are then stored in the VRF routing table for that CE router.

The PE router on one side of the non-MPLS network is learned by the PE router on the other side of the non-MPLS network though a routing protocol running within the non-MPLS network. Routes between the PE routers are stored in the main or default routing table.

Routes of the customer networks behind the PE router are learned by the other PE router through BGP and are not known to the non-MPLS network. This is accomplished by defining a static route to the BGP neighbor (the other PE router) through a GRE tunnel across the non-MPLS network. When routes are learned from the BGP neighbor, they will have the next-hop of the GRE tunnel and thus all customer network traffic will be sent using the GRE tunnel.

GRE Tunnel Attached to a Cisco 7600 SIP-400 Interface or Subinterface

For the Cisco 7600 series router to perform the MPLS and mVPN processing and have the Cisco 7600 SIP-400 perform the GRE processing, interfaces or subinterfaces must have an IP address. The MPLS and protocol independent multicast (PIM) configuration must be on the tunnel interface. The Cisco 7600 series router views the Cisco 7600 SIP-400 main interface or subinterface as an MPLS or PIM interface, so MPLS and mVPN processing is performed, and provides the Cisco 7600 SIP-400 with the correlation information needed to perform GRE processing.

Tunnel Interface Configuration

The **ip pim sparse-mode** command must be configured on the tunnel interface. It should not be configured on the physical interface or subinterface facing core. It is automatically configured on the Cisco 7600 SIP-400 interface or subinterface when a tunnel is attached to the interface or subinterface. The tunnel source IP address is typically a lookback address.

Displaying Unicast Routes

The display of unicast routes (Main Routing Table) shows the next hop for the BGP neighbor to be the Cisco 7600 SIP-400 interface or subinterface. On a router that natively supports this feature, the next hop for the BGP neighbor is the tunnel interface.

The following example shows the output from the **show ip route** command:

```
router# show ip route | inc Tunnel
S       4.4.4.4 is directly connected, Tunnel0
C       1.0.0.0 is directly connected, Tunnel0
```

Displaying Multicast Routes

The display of multicast routes (groups) shows the output interface for the 239.0.0.0/8 group to be the Cisco 7600 SIP-400 interface or subinterface. On a router that natively supports this feature, the output interface is the tunnel interface.

The following example shows the output from the **show ip mroute** command:

```
router# show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group
       V - RD & Vector, v - Vector
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 01:23:02/00:03:22, RP 2.2.2.2, flags: SJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel0, Forward/Sparse-Dense, 00:03:45/00:03:22
    Loopback0, Forward/Sparse-Dense, 01:23:02/00:02:30

(*, 239.1.1.2), 01:23:01/00:02:35, RP 2.2.2.2, flags: SJCZ
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel0, Forward/Sparse-Dense, 00:03:45/00:02:34
    MVRF vpn1, Forward/Sparse-Dense, 01:23:01/00:02:12

(2.2.2.2, 239.1.1.2), 01:22:50/00:03:29, flags: T
  Incoming interface: Loopback0, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    Tunnel0, Forward/Sparse-Dense, 00:03:45/00:02:54, H

(4.4.4.4, 239.1.1.2), 00:03:33/00:02:59, flags: TZ
  Incoming interface: Tunnel0, RPF nbr 1.0.0.2, RPF-MFD
  Outgoing interface list:
    MVRF vpn1, Forward/Sparse-Dense, 00:03:33/00:02:26, H

(*, 239.1.1.1), 01:23:01/stopped, RP 2.2.2.2, flags: SJCZ
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    MVRF vpn3, Forward/Sparse-Dense, 01:23:01/00:02:11

(2.2.2.2, 239.1.1.1), 01:22:50/00:02:59, flags: PT
  Incoming interface: Loopback0, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list: Null

router# show ip mroute vrf vpn1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group
       V - RD & Vector, v - Vector
Outgoing interface flags: H - Hardware switched, A - Assert winner
```

```
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 01:23:11/00:02:24, RP 200.200.200.200, flags: SJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Loopback200, Forward/Sparse-Dense, 01:23:11/00:02:24
    Tunnel16, Forward/Sparse-Dense, 00:03:40/00:02:32

(*, 224.1.2.3), 00:02:43/stopped, RP 200.200.200.200, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel16, Forward/Sparse-Dense, 00:02:43/00:02:43

(100.0.1.2, 224.1.2.3), 00:00:17/00:03:20, flags: T
  Incoming interface: GigabitEthernet2/0/0.1, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    Tunnel16, Forward/Sparse-Dense, 00:00:17/00:03:12, H

(*, 224.1.2.2), 00:02:43/stopped, RP 200.200.200.200, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel16, Forward/Sparse-Dense, 00:02:44/00:02:42

(100.0.1.2, 224.1.2.2), 00:00:18/00:03:20, flags: T
  Incoming interface: GigabitEthernet2/0/0.1, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    Tunnel16, Forward/Sparse-Dense, 00:00:18/00:03:11, H

(*, 224.1.2.1), 00:02:44/stopped, RP 200.200.200.200, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel16, Forward/Sparse-Dense, 00:02:44/00:02:44

(100.0.1.2, 224.1.2.1), 00:00:19/00:03:19, flags: T
  Incoming interface: GigabitEthernet2/0/0.1, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    Tunnel16, Forward/Sparse-Dense, 00:00:19/00:03:10, H
```

Displaying Tunnel-to-Interface Mappings

The **show cwan mplsogre** command displays the tunnel-to-interface mappings. The following example illustrates the output of the **show cwan mplsogre** command:

```
Router# show cwan mplsogre
gigabitethernet 2/0/0
  Tunnel1 is attached
    Interface
      VLAN: 1022, STATE: UP
      IP Address: 6.0.0.1          IP Mask: 255.0.0.0
    Tunnel
      VLAN: 1017, STATE: UP
      IP Address: 8.0.0.1          IP Mask: 255.0.0.0
      Src Address: 6.0.0.1, Dst Address: 7.0.0.1
      Static Routes to Tunnel: 1
        IP Address: 4.0.0.1          IP Mask: 255.255.255.255
```

Scalable EoMPLS

In Cisco IOS Release 12.2(33)SRA and later, Scalable EoMPLS now allows a Cisco 7600 SIP-400-based line card to face the CE. This configuration allows the platform to scale the number of EoMPLS virtual connections (VCs) that it can support from 4K to 12K. When AToM xconnect commands are placed on Cisco 7600 SIP-400 subinterfaces, the line card performs AToM imposition and disposition. Supervisor hardware will perform only MPLS switching on traffic from these interfaces. Additionally, configuring xconnect commands on Cisco 7600 SIP-400 subinterfaces will not consume globally significant VLANs on a per-xconnect basis. This change also provides the ability to support FRR on EoMPLS VCs with the same model as other CEF/MFI-based AToM configurations.

To achieve this scalability, Cisco 7600 SIP-400 must be the CE facing line card as opposed to the current model of a LAN line card facing the CE. With Cisco 7600 SIP-400 configured for Scalable EoMPLS, any line card capable of switching MPLS packets may be core facing.

On a Sup720 system, configuring EoMPLS under a non-VLAN interface is considered hardware-based EoMPLS. Configuring EoMPLS on a VLAN interface is considered to be software-based MPLS. Configuring EoMPLS on Cisco 7600 SIP-400 subinterfaces is considered to be Scalable EoMPLS.

Configuring Flow Control Support on the Link

Flow control is turned on or off based on the result of autonegotiation. Flow control is not supported on the Cisco 7600 SIP-200 and Cisco 7600 SIP-400, so it will always negotiate to off. Flow control can be configured independently of autonegotiation on the Cisco 7600 SIP-600. For information on this process, see the "Configuring Autonegotiation on an Interface" section on page 12-11.

This section discusses the following topics:

- Verifying Flow Control Status for an Ethernet SPA Interface on a Cisco 7600 SIP-200, page 12-21
- Verifying Flow Control Status for a Gigabit Ethernet SPA Interface on a Cisco 7600 SIP-400, page 12-22
- Configuring Flow Control for an Ethernet SPA Interface on a Cisco 7600 SIP-600, page 12-22

Verifying Flow Control Status for an Ethernet SPA Interface on a Cisco 7600 SIP-200

The following example shows how to verify that flow control pause frames are being transmitted and received for a Fast Ethernet SPA on the Cisco 7600 SIP-200.

```
Router# show hw sub 2 counter mac
    Show counters info for Subslot 2:
    port:0
    good_octets_received: 2046026640038
    bad_octets_received: 0
    good_frames_received: 31969140675
    bad_frames_received: 0
    broadcast_frames_received: 2
    multicast_frames_received: 3562
    good_octets_sent: 1373554315151
    good_frames_sent: 22892514199
    broadcast_frames_sent: 0
    multicast_frames_sent: 0
    mac_transfer_error: 0
    excessive_collision: 0
    unrecog_mac_control_received: 0
    fc_sent: 11232431
    good_fc_received: 0
```

```
rx_over_flow_events: 234082101
undersize: 0
fragments: 0
oversize: 0
jabber: 0
mac_rcv_error: 0
bad_crc: 0
collisions: 0
late_collision: 0
rate_limit_dropped: 0
tx_fifo_full_packet_drops : 0
spi4_rx_frames: 2814271686
spi4_tx_frames: 1328805298
```

Verifying Flow Control Status for a Gigabit Ethernet SPA Interface on a Cisco 7600 SIP-400

To verify flow control status on a Gigabit Ethernet interface on a SPA, use the **show interfaces gigabitethernet** privileged EXEC command and view the "output flow-control is" and "input flow-control is" output lines to see if input and output flow control is on or off. The "pause input" and "pause output" counters of the output of this command can be used to view the number of pause frames sent or received by the interface.

The following example shows that zero pause frames have been transmitted and received by the MAC device for interface port 1 (the second port) on the SPA located in subslot 0 of the SIP that is installed in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces gigabitethernet 2/0/1
GigabitEthernet2/0/1 is up, line protocol is up
  Hardware is GigEther SPA, address is 000a.f330.2e40 (bia 000a.f330.2e40)
  Internet address is 2.2.2.1/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
  Full-duplex, 1000Mb/s, link type is force-up, media type is SX
 output flow-control is off, input flow-control is off
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 03:18:49, output 03:18:44, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     1703 packets input, 638959 bytes, 0 no buffer
     Received 23 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 1670 multicast, 0 pause input
     1715 packets output, 656528 bytes, 0 underruns
     0 output errors, 0 collisions, 4 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 pause output
     0 output buffer failures, 0 output buffers swapped out
```

Configuring Flow Control for an Ethernet SPA Interface on a Cisco 7600 SIP-600

On the Cisco 7600 SIP-600, flow control can be configured on Ethernet SPA interfaces by entering the **flowcontrol send** command to configure the interface to transmit pause frames or the **flowcontrol receive** command to configure the interface to receive pause frames.

Command	Purpose
Router(config-if)# **flowcontrol send** [**desired** | **off** | **on**]	Enables transmission of outgoing pause frames. The following options can be configured with this command: • **desired**—Allows, but does not require, outgoing pause frames to leave the interface. • **off**—Disables transmission of outgoing pause frames. • **on**—Enables transmission of outgoing pause frames.
Router(config-if)# **flowcontrol receive** [**desired** | **off** | **on**]	Enables the interface to receive incoming pause frames. The following options can be configured with this command: • **desired**—Allows, but does not require, the interface to receive incoming pause frames. • **off**—Does not allow incoming pause frames to enter the interface. • **on**—Allows incoming pause frames to enter the interface.

Note When a user configures flow control for either the transmit or receive direction, it is automatically enabled for both transmit and receive directions simultaneously.

Fast Ethernet SPAs have flow control enabled by default and it cannot be disabled.

Configuring EtherChannels

An EtherChannel bundles individual Ethernet links into a single logical link that provides the aggregate bandwidth of up to eight physical links.

Note EtherChannel is only supported on the 10-Port Gigabit Ethernet SPA and the 1-Port 10-Gigabit Ethernet SPA on the Cisco 7600 SIP-600. EtherChannel is not supported on the 2-Port Gigabit Ethernet SPA on the Cisco 7600 SIP-400 or on a Fast Ethernet SPA on the Cisco 7600 SIP-200.

For additional information on EtherChannels, see the "Configuring EtherChannels" section in the "Configuring Layer 3 and Layer 2 EtherChannel" chapter of the *Cisco 7600 Series Router Cisco IOS Software Configuration Guide* publication that corresponds to your Cisco IOS software release.

Configuring Virtual Private LAN Service (VPLS) and Hierarchical VPLS

VPLS enables geographically separate LAN segments to be interconnected as a single bridged domain over a packet switched network, such as IP, MPLS, or a hybrid of both bridging techniques.

VPLS with EoMPLS uses an MPLS-based provider core, where the PE routers have to cooperate to forward customer Ethernet traffic for a given VPLS instance in the core. VPLS uses the provider core to join multiple attachment circuits together to simulate a virtual bridge that connects the multiple attachment circuits together. From a customer point of view, there is no topology for VPLS. All of the CE devices appear to connect to a logical bridge emulated by the provider core.

For VPLS and hierarchical virtual private LAN service (H-VPLS) configuration guidelines, restrictions, and feature compatibility tables, see the "Configuring Virtual Private LAN Service" section on page 4-46.

Note H-VPLS is not available on Fast Ethernet SPAs.

Configuring Connectivity Fault Management

Ethernet Connectivity Fault Management (CFM) is an end-to-end per-service-instance Ethernet layer Operation, Administration, and Management (OAM) protocol. It includes proactive connectivity monitoring, fault verification, and fault isolation for large Ethernet metropolitan-area and wide-area networks (MANs and WANs). See the *Ethernet Connectivity Fault Management* document for more information on this feature.

Ethernet CFM Configuration Guidelines

The following apply to the Cisco 7600 SIP-400:

- Ethernet SPAs cannot be configured as switchport.
- Multiple domains with the same level can be configured, that is, different domain names at the same maintenance level. Associating a single domain name with multiple maintenance levels is not supported.
- The user must first configure the MIP level on an interface before configuring inward Maintenance End Points (MEPs) on that interface. The exceptions to this are when configuring inward MEPs of Level 7 or when configuring outward facing MEPs, which are allowed without having to configure any MIPs.
- Routed (Layer 3) ports may only have outward facing MEPs; no MIPs are allowed. MIP configuration on a routed port will be rejected and an error message will be generated.
- Note that all MEPs and MIPs must be removed from a port before MEPs of Level 7 can be configured. Also, when unconfiguring MEPs of Level 7, the user should remove any lower level MEPs first.
- Configuring a MEP on an interface with a level higher than the MIP level will generate an error message.
- A single interface may belong to multiple domains, configuring multiple instances of the **ethernet cfm mep level** command for different domains is supported.
- A specified domain must be configured or an error message will be displayed.
- If an interface is provisioned to be a MIP for a certain maintenance level, and MEP is configured for a VLAN on the same level, an error message will be displayed.
- When specifying an outward MEP, the domain-name must be provided. If the specified domain has not been configured or if the specified domain has not been tagged as outward, an error message will be displayed.

- If the VLAN for which a MEP is configured gets removed from an interface, the MEP configuration will be removed as well, since the definition of the MEP is tied to the VLAN.

Configuring Maintenance Domains and Maintenance Points

This section describes configuring maintenance domains and maintenance points.

Configuring the Ethernet Domain

Use the following command to configure the Ethernet domain:

Command	Purpose
Router(config)# **ethernet cfm domain** *domain-name* **level** {*0* to *7*} [**direction outward**]	Defines a CFM Maintenance domain at a particular maintenance level. It sets the router into config-ether-cfm configuration mode, where parameters specific to the maintenance domain can be set. • **Direction outward** (optional)—Specifies the domain direction. Specifying a domain as outward allows for the creation of multiple outward domains at the same level containing an overlapping set of vlans. The set of vlans in an outward domain can also overlap with inward domains. Note that the set of vlans between inward domains at the same level must still be unique.

Configuring Maintenance Points

To set a port as internal to a maintenance domain and define it as a MEP, use the following command:

\

Command	Purpose
Router(config-interface)# **ethernet cfm mep level** {*0* to *7*} {**inward** | **outward** *domain-name*} **mpid** *id* **vlan** {*vlan-id* | *any* | *vlan-id-vlan-id* [*vlan-id-vlan-id*}	• **inward** | **outward**—Indicates the direction of the MEP as either **inward** (towards the bridge) or **outward** (towards the wire). The default is inward facing. • *domain-name*—A string of maximum length of 256 characters. • *id*—A string of maximum length of 256 characters. • *vlan-id*—An integer from 1 to 4095. **Note** A comma must be entered to separate each VLAN ID range from the next range. **Note** Hyphen must be entered to separate the starting and ending VLAN ID values that are used to define a range of VLAN IDs.

Verifying Ethernet CFM Configuration

The following commands can be used to verify CFM configuration:

Command	Purpose
Router# **show ethernet cfm maintenance-points local** [**mep** | **mip**] [**interface** *interface-name* | **domain** *domain-name* | **level** {*0* to *7*}]	Displays the local maintenance points configured on the device. Allows filtering of output as follows: • Displays all maintenance points independent of domain or interface. • Displays all maintenance points on a particular interface independent of domain • Displays all maintenance points on a particular interface belonging to a given domain • Displays all maintenance points belonging to a given domain independent of interface The display may also be restricted to either MEPs or MIPs. • *domain-name*— (optional) A string of maximum length of 256 characters.

The **show ethernet cfm maintenance-points local** displays the local maintenance points that are configured:

```
Router# show ethernet cfm maintenance-points local
MPID DomainName           Level Type  VLAN Port       CC-Status MAC
1522 DOMAIN_PROVIDER_L5_1 5     MEP I 2    Et2/0.1    Enabled   aabb.cc00.0100
1502 DOMAIN_PROVIDER_L5_1 5     MEP O 2    Et0/0.1    Enabled   aabb.cc00.0100
1523 DOMAIN_PROVIDER_L5_1 5     MEP O 3    Et2/0.2    Enabled   aabb.cc00.0100
1503 DOMAIN_PROVIDER_L5_1 5     MEP I 3    Et0/0.2    Enabled   aabb.cc00.0100
1302 DOMAIN_OPERATOR_L3_1 3     MEP I 2    Et0/0.1    Enabled   aabb.cc00.0100
1303 DOMAIN_OPERATOR_L3_1 3     MEP I 3    Et0/0.2    Enabled   aabb.cc00.0100

Level Type  Port                       MAC
7     MIP   Et2/0.2                    aabb.cc00.0100
7     MIP   Et2/0.1                    aabb.cc00.0100
7     MIP   Et0/0.2                    aabb.cc00.0100
7     MIP   Et0/0.1                    aabb.cc00.0100
```

Command	Purpose
Router# **ping ethernet** <*mac-address*> {**domain** *domain-name* | **level** {0 to 7}} **vlan** *vlan-id* [**source** *mpid*]	Sends Ethernet CFM loopback messages to the destination MAC address. • *mac-address*—MAC Address of remote maintenance point, in the format abcd.abcd.abcd. • *domain-name*—A string of maximum. length of 256 characters. • v*lan-id*—An integer from 1 to 4095.

The **ping ethernet** command shows loopback messages on the destination MAC address:

```
Router# ping ethernet
Sending 5, 100-byte Ethernet CFM Echoes to <mac-address>, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/4 ms
```

Configuring Ethernet Operations, Administration, and Maintenance

The Gigabit Ethernet SPAs support OAM as defined by IEEE 802.3ah, *Ethernet in the First Mile*. IEEE 802.3ah operates on a single point-to-point link between two devices using slow protocol packets called OAM protocol data units (OAMPDUs) that are never forwarded.

IEEE 802.3ah defines five functional areas, of which the Gigabit Ethernet SPAs on the Cisco 7600 series router support the following three:

- OAM discovery—Supports identification of OAM support and capabilities on a peer device.
- Link monitoring—Provides event notification and link information. It also supports polling and response (but not writing) of the 802.3ah MIB.
- Remote failure indication—Supports informing a peer device that the receive path is down. This requires support of unidirectional operation on the link.

Ethernet OAM Configuration Guidelines

When configuring Ethernet OAM on the SPAs, consider the following guidelines:

- See Table 12-2 for information about where the OAM features for SPA interfaces are supported.
- On Gigabit Ethernet links, the unidirectional fault signaling support in OAM and the autonegotiation capabilities of Gigabit Ethernet (IEEE 802.3z) are mutually exclusive. You must disable autonegotiation for OAM fault signaling to be sent over unidirectional links.
- Ethernet OAM requires point-to-point links where OAMPDUs are created and terminated.
- When configuring Ethernet OAM interface modes, consider the following guidelines:
  - At least one of the peer interfaces must be in active mode.
  - The peer interfaces either can be both in active mode, or one can be in active mode and the other in passive mode.
  - You can change Ethernet OAM modes without disabling OAM.

- When using templates to configure Ethernet OAM interfaces, consider the following guidelines:
  - If you use a template to configure common or global OAM characteristics and apply it an interface, you can override any of the configuration statements in the template by configuring the same command at the interface with a different value.
  - You can define multiple templates to create different sets of link monitoring characteristics.
  - You can only apply one template to any single Ethernet OAM interface.

Table 12-2 provides information about where the OAM features for SPA interfaces are supported.

Table 12-2 Ethernet OAM Feature Compatibility by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
• OAM discovery • Link monitoring • Remote failure indication (Dying Gasp only)	Not supported.	In Cisco IOS Release 12.2(33)SRA: • 2-Port Gigabit Ethernet SPA	In Cisco IOS Release 12.2(33)SRA: • 1-Port 10-Gigabit Ethernet SPA • 5-Port Gigabit Ethernet SPA • 10-Port Gigabit Ethernet SPA
Remote loopback	Not supported.	Not supported.	Not supported.
MIB variable retrieval	Not supported.	Not supported.	Not supported.

Ethernet OAM Configuration Tasks

The following sections describe the Ethernet OAM configuration tasks:

- Enabling OAM on an Interface, page 12-28 (required)
- Enabling and Disabling a Link Monitoring Session, page 12-30 (optional)
- Starting and Stopping Link Monitoring Operation, page 12-30 (optional)
- Configuring Link Monitoring Options, page 12-31 (optional)
- Configuring Remote Failure Indication Actions, page 12-38 (optional)
- Configuring Global Ethernet OAM Options Using a Template, page 12-39 (optional)
- Verifying Ethernet OAM Configuration, page 12-40

Enabling OAM on an Interface

OAM is disabled on an interface by default. When you enable OAM on an interface, the interface automatically advertises to the remote peer that it supports link monitoring during OAM discovery. Link monitoring support must be agreed upon by the peer interfaces for monitoring to operate across the link.

Once link monitoring support is achieved between the peer interfaces, the interface will start the link monitoring operation, and send event OAMPDUs when errors occur locally, and interpret event OAM PDUs received by the remote peer.

You do not need to explicitly configure link monitoring support, or start the link monitoring operation on the link unless you have previously disabled monitoring support or operation on the interface.

To enable OAM features on a Gigabit Ethernet interface, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface** *type slot/subslot/port*	Specifies the Ethernet SPA interface, where • *type*—Specifies the type of Ethernet interface, such as **gigabitethernet** or **tengigabitethernet**. • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4. **Note** Ethernet OAM can be defined on a main Gigabit Ethernet interface only—not on subinterfaces.
Step 2	Router(config-if)# **ethernet oam** [**max-rate** *oampdus* | **min-rate** *num-seconds* | **mode** {**active** | **passive**} | **timeout** *seconds*]	Enables OAM on a Gigabit Ethernet interface, where: • **max-rate** *oampdus*—(Optional) Specifies the maximum number of OAMPDUs that can be sent per second as an integer in the range of 1 to 10. The default is 10. • **min-rate** *num-seconds*—(Optional) Specifies the number of seconds (in the range 1–10) during which at least one OAMPDU must be sent. The default is 1 second. • **mode** {**active** | **passive**}—(Optional) Specifies the client mode for OAM discovery and link negotiation, where: – **active**— Specifies that the interface initiates OAMPDUs for protocol negotiation as soon as the interface becomes active. This is the default. At least one of the OAM peers must be configured in active mode. – **passive**—Specifies that the interface waits in a listening mode to receive an incoming OAMPDU for protocol negotiation from a peer. The passive interface begins sending OAMPDUs once it receives OAMPDUs from the peer.

Command	Purpose
	Note If you configure an interface in passive mode, then you must be sure that the peer is in active mode for successful OAM operation. • **timeout** *seconds*—Specifies the amount of time, in seconds (in the range 2–30), after which a device declares its OAM peer to be nonoperational and resets its state machine. The default is 5 seconds.

Enabling and Disabling a Link Monitoring Session

The OAM peer interfaces must establish a link monitoring session before the actual operation of link monitoring can begin. If you have enabled OAM on the interface, and have not explicitly disabled link monitoring support on the interface, then you do not need to explicitly configure link monitoring support on the interface to establish a session.

The **ethernet oam link-monitor supported** command automatically runs in the background when you configure the **ethernet oam** interface configuration command. Be sure that at least one of the Ethernet OAM peers is configured for active mode so that a session can be established.

To explicitly configure and enable a link monitoring session on an interface, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor supported**	Enables link monitoring support on an Ethernet OAM interface.

To disable a link monitoring session on an interface, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **no ethernet oam link-monitor supported**	Disables link monitoring support on an Ethernet OAM interface.

Starting and Stopping Link Monitoring Operation

If a link monitoring session is established among the Ethernet OAM peer interfaces, then sending and receiving of Event Notification OAMPDUs can begin between the peers. This link monitoring operation across the link automatically starts when you enable OAM on the interface.

The **ethernet oam link-monitor on** command automatically runs in the background when you configure the **ethernet oam** interface configuration command.

You can stop and restart the operation of link monitoring (or, the sending and receiving of Event Notification OAMPDUs on a link). Stopping link monitoring operation is not the same thing as disabling link monitoring support. When you stop link monitoring operation, the interface is still configured to support link monitoring with its peer, but just is not actively sending and receiving Event Notification OAMPDUs.

To explicitly configure and start link monitoring operation on an interface, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor on**	Starts link monitoring on an Ethernet OAM interface.

To stop link monitoring operation on an interface, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **no ethernet oam link-monitor on**	Stops link monitoring on an Ethernet OAM interface.

Configuring Link Monitoring Options

When OAM link monitoring is active, Event Notification OAMPDUs are sent to a remote OAM client when errors are detected locally. You can configure certain windows and thresholds to define when these error event notifications are triggered. If you do not modify the link monitoring options, default values are used for the window periods and low thresholds.

The Gigabit Ethernet SPAs support the following types of error events as defined by IEEE 802.3ah:

- Errored Symbol Period (errored symbols per second)—This event occurs when the number of symbol errors during a specified period exceeds a threshold. These are coding symbol errors (for example, a violation of 4B/5B coding).
- Errored Frame (errored frames per second)—This event occurs when the number of frame errors during a specified period exceeds a threshold.
- Errored Frame Period (errored frames per N frames)—This event occurs when the number of frame errors within the last N frames exceeds a threshold.
- Errored Frame Seconds Summary (errored seconds per M seconds)—This event occurs when the number of errored seconds (one second intervals with at least one frame error) within the last M seconds exceeds a threshold.

Cisco Systems adds the following types of vendor-specific error events:

- Receive CRC (errored frames per second)—This event occurs when the number of frames received with CRC errors during a specified period exceeds a threshold.
- Transmit CRC (errored frames per second)—This event occurs when the number of frames transmitted with CRC errors during a specified period exceeds a threshold.

The link monitoring options can be configured in a global template that can be applied to one or more interfaces, and also can be explicitly configured at the interface.

Specifying Errored Symbol Period Link Monitoring Options

The errored symbol period link monitoring options include the ability to specify the number of symbols to be tracked or counted for errors, and the high and low thresholds for triggering the Errored Symbol Period Link Event.

To specify errored symbol period link monitoring options, use the following commands in interface configuration or template configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor symbol-period window** *million-symbol-units*	(Optional) Specifies the number of symbols (in the range 1–65535, as a multiple of 1 million symbols) to be included in the error counting according to the specified thresholds. The default window unit is 100, or 100 million symbols.
Router(config-if)# **ethernet oam link-monitor symbol-period threshold low** *low-symbols*	(Optional) Specifies the low errored symbol threshold as a number of symbol errors (in the range 0–65535). If the number of error symbols in the window period is equal to or greater than *low-symbols*, then the Errored Symbol Period Link Event will be generated. The default low threshold is 0 symbols.
Router(config-if)# **ethernet oam link-monitor symbol-period threshold high** {**none** | *high-symbols*}	(Optional) Specifies the high errored symbol threshold as a number of error symbols (in the range 1–65535). If the number of error symbols in the window period is equal to or greater than *high-symbols*, then a user defined action will be triggered. There is no default for the high threshold, so you must explicitly configure a value to enable it. For more information about configuring a user-defined action, see "Specifying a High Threshold Action" section on page 12-37.

Specifying Errored Frame Link Monitoring Options

The errored frame link monitoring options include the ability to specify a period of time during which frame errors are tracked or counted, and the high and low thresholds for triggering the Errored Frame Link Event. The Gigabit Ethernet SPAs on the Cisco 7600 series router count general frame errors, such as CRC errors and corrupted packets, as errored frames.

To specify errored frame link monitoring options, use the following commands in interface configuration or template configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor frame window** *100-millisecond-units*	(Optional) Specifies a period of time (in the range 10–600, as a multiple of 100 milliseconds) during which error counting occurs according to the specified thresholds. The default window unit is 10, or 1000 milliseconds.

Command	Purpose
Router(config-if)# **ethernet oam link-monitor frame threshold low** *low-frames*	(Optional) Specifies the low error frame threshold as a number of frames (in the range 0–65535). If the number of error frames in the window period is equal to or greater than *low-frames*, then the Errored Frame Link Event will be generated. The default low threshold is 0 frame errors.
Router(config-if)# **ethernet oam link-monitor frame threshold high** {**none** | *high-frames*}	(Optional) Specifies the high error frame threshold as a number of error frames (in the range 1–65535). If the number of error frames in the window period is equal to or greater than *high-frames*, then a user defined action will be triggered. There is no default for the high threshold, so you must explicitly configure a value to enable it. Use the **none** keyword to disable the high threshold. For more information about configuring a user-defined action, see "Specifying a High Threshold Action" section on page 12-37.

Specifying Errored Frame Period Link Monitoring Options

The errored frame period link monitoring options include the ability to specify the number of error frames to be tracked or counted for errors, and the high and low thresholds for triggering the Errored Frame Period Link Event. The Gigabit Ethernet SPAs on the Cisco 7600 series router count general frame errors, such as CRC errors and corrupted packets, as errored frames.

To specify errored frame period link monitoring options, use the following commands in interface configuration or template configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor frame-period window** *10000-frame-units*	(Optional) Specifies the number of frames (in the range 1000–65535, as a multiple of 10000 frames) to be included in the error counting according to the specified thresholds. The default window unit is 1000, or 10000000 frames.

Command	Purpose
Router(config-if)# **ethernet oam link-monitor frame-period threshold low** *low-frames*	(Optional) Specifies the low error frame threshold as a number of frames (in the range 0–65535). If the number of error frames in the window period is equal to or greater than *low-frames*, then the Errored Frame Period Link Event will be generated. The default low threshold is 0 frame errors.
Router(config-if)# **ethernet oam link-monitor frame-period threshold high** {**none** | *high-frames*}	(Optional) Specifies the high error frame threshold as a number of frames (in the range 1–65535). If the number of error frames in the window period is equal to or greater than *high-frames*, a user defined action will be triggered. There is no default for the high threshold, so you must explicitly configure a value to enable it. Use the **none** keyword to disable the high threshold. For more information about configuring a user-defined action, see "Specifying a High Threshold Action" section on page 12-37.

Specifying Errored Frame Seconds Summary Link Monitoring Options

The errored frame seconds summary link monitoring options include the ability to specify a period of time during which tracking of a number of errored-seconds periods (one-second intervals with at least one frame error) occurs, and the high and low thresholds for triggering the Errored Frames Seconds Summary Link Event.

To specify errored frame seconds summary link monitoring options, use the following commands in interface configuration or template configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor frame-seconds window** *100-millisecond-units*	(Optional) Specifies a period of time (in the range 100–9000, as a multiple of 100 milliseconds) during which tracking of an errored-seconds period occurs according to the specified thresholds. The default window unit is 100, or 10000 milliseconds.

Command	Purpose
Router(config-if)# **ethernet oam link-monitor frame-seconds threshold low** *low-errored-seconds*	(Optional) Specifies the low errored seconds threshold as a number of errored seconds (in the range 0–900). If the number of errored seconds in the window period is equal to or greater than *low-errored-seconds*, then the Errored Frame Seconds Summary Link Event will be generated. The default low threshold is 0 error seconds.
Router(config-if)# **ethernet oam link-monitor frame-seconds threshold high** {**none** | *high-errored-seconds*}	(Optional) Specifies the high errored seconds threshold as a number of errored seconds (in the range 1–900). If the number of errored seconds in the window period is equal to or greater than *high-errored-seconds*, then a user defined action will be triggered. There is no default for the high threshold, so you must explicitly configure a value to enable it. Use the **none** keyword to disable the high threshold. For more information about configuring a user-defined action, see "Specifying a High Threshold Action" section on page 12-37.

Specifying Receive CRC Link Monitoring Options

The receive CRC link monitoring options include the ability to specify a period of time during which tracking of frames received with CRC occurs, and the high and low thresholds for triggering the error. Receive CRC link monitoring is a Cisco-specific implementation and is only locally significant to the Ethernet OAM interface on the Cisco 7600 series router.

To specify receive CRC link monitoring options, use the following commands in interface configuration or template configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor receive-crc window** *100-millisecond-units*	(Optional) Specifies a period of time (in the range 10–1800, as a multiple of 100 milliseconds) during which tracking of frames received with CRC errors occurs according to the specified thresholds. The default window unit is 10, or 1000 milliseconds.

Command	Purpose
Router(config-if)# **ethernet oam link-monitor receive-crc threshold low** *low-frames*	(Optional) Specifies the low CRC threshold as a number of frames (in the range 0–65535). If the number of frames received with CRC errors in the window period is equal to or greater than *low-frames*, then the Receive CRC error will be generated. The default low threshold is 1 frame.
Router(config-if)# **ethernet oam link-monitor receive-crc threshold high** {**none** | *high-frames*}	(Optional) Specifies the high CRC threshold as a number of frames (in the range 1–65535). If the number of frames received with CRC errors in the window period is equal to or greater than *high-frames*, a user defined action will be triggered. There is no default for the high threshold, so you must explicitly configure a value to enable it. Use the **none** keyword to disable the high threshold. For more information about configuring a user-defined action, see "Specifying a High Threshold Action" section on page 12-37.

Specifying Transmit CRC Link Monitoring Options

The transmit CRC link monitoring options include the ability to specify a period of time during which tracking of frames transmitted with CRC occurs, and the high and low thresholds for triggering the error. Transmit CRC link monitoring is a Cisco-specific error event and is only locally significant to the Ethernet OAM interface on the Cisco 7600 series router.

To specify transmit CRC link monitoring options, use the following commands in interface configuration or template configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor transmit-crc window** *100-millisecond-units*	(Optional) Specifies a period of time (in the range 10–1800, as a multiple of 100 milliseconds) during which tracking of frames received with CRC errors occurs according to the specified thresholds. The default window unit is 10, or 1000 milliseconds.

Command	Purpose
Router(config-if)# **ethernet oam link-monitor transmit-crc threshold low** *low-frames*	(Optional) Specifies the low CRC threshold as a number of frames (in the range 0–65535). If the number of frames transmitted with CRC errors in the window period is equal to or greater than *low-frames*, then the Receive CRC error will be generated. The default low threshold is 1 frame.
Router(config-if)# **ethernet oam link-monitor transmit-crc threshold high** {**none** | *high-frames*}	(Optional) Specifies the high CRC threshold as a number of frames (in the range 1–65535). If the number of frames transmitted with CRC errors in the window period is equal to or greater than *high-frames*, a user defined action will be triggered. There is no default for the high threshold, so you must explicitly configure a value to enable it. Use the **none** keyword to disable the high threshold. For more information about configuring a user-defined action, see "Specifying a High Threshold Action" section on page 12-37.

Specifying a High Threshold Action

When you configure high thresholds for OAM link monitoring, you can specify an action to be taken when the high threshold is exceeded.

When configuring high threshold actions, consider the following guidelines:

- There is no default action.
- If you configure a high threshold but do not configure any corresponding action, only a message appears on the syslog and no other action is taken on the interface.
- If you want to associate different high threshold actions for different kinds of link monitoring functions, you can use configuration templates. However, only one configuration template can be applied to any Ethernet OAM interface.
- Only one high threshold action can be configured for any Ethernet OAM interface.

To configure an action when a high threshold for an error is exceeded on an Ethernet OAM interface, use the following command in interface configuration or template configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam link-monitor high-threshold action** {**error-disable-interface** | **failover**}	(Optional) Configures the action when a high threshold error is exceeded, where: • **error-disable-interface**—Shuts down the Ethernet OAM interface. • **failover**—(EtherChannel interface only) Configures the interface for an automatic failover of traffic from one port in an EtherChannel to another port in the same EtherChannel when one of the ports in the channel exceeds the high error threshold within the specified interval. The port failover only occurs if there is at least one operational port available in the EtherChannel. The failed port will be put into an error disable state. If the failed port is the last port in the EtherChannel, the port will not be put into an error disable state and continues to pass traffic regardless of the type of errors being received. Single, nonchanneling ports go into the error disable state when the error threshold is exceeded within the specified interval.

Configuring Remote Failure Indication Actions

When an RFI event occurs locally, the local client sends an Information OAMPDU to its peer with a bit selected that indicates the type of failure. The Gigabit Ethernet SPAs on the Cisco 7600 series router process all of the following types of Remote Failure Indication (RFI) conditions as defined by IEEE 802.3ah:

- Critical Event—This type of RFI is sent when an unspecified critical event has occurred. These events are vendor specific, and the failure indication might be sent immediately and continuously.
- Dying Gasp—This type of RFI is sent when an unrecoverable condition (for example, a power failure) has occurred. The conditions for a dying gasp RFI are vendor specific, and the failure indication might be sent immediately and continuously. The Gigabit Ethernet SPAs on the Cisco 7600 series router generate a Dying Gasp RFI when an interface is error-disabled or administratively shut down. This is the only type of RFI that the Gigabit Ethernet SPAs on the Cisco 7600 series router generate.
- Link Fault—This type of RFI is sent when a loss of signal is detected by the receiver (for example, a peer's laser is malfunctioning). A link fault is sent once per second in the Information OAMPDU. The link fault RFI applies only when the physical sublayer is capable of independent transmit and receive.

When the Gigabit Ethernet SPAs receive an OAMPDU with an RFI bit selected, a syslog message is created providing the failure reason, as shown in the following example:

```
%ETHERNET_OAM-SP-6-RFI: The client on interface Gi1/1 has received a remote failure
indication from its remote peer (failure reason = remote client administratively turned
off)
```

You can configure a response, or action, by the local client to shut down the OAM interface when it receives Information OAMPDUs with a Dying Gasp RFI bit selected.

To configure an error disable action for the local Ethernet OAM interface, use the following command in interface configuration or template configuration mode:

Command	Purpose
Router(config-if)# **ethernet oam remote-failure dying-gasp action error-disable-interface**	(Optional) Specifies that the local Ethernet OAM interface is shut down upon receipt of an Information OAMPDU from its peer that indicates a Dying Gasp.

Configuring Global Ethernet OAM Options Using a Template

Create configuration templates when you have a common set of link-monitoring or remote-failure characteristics that you want to apply to multiple Ethernet OAM interfaces. This streamlines Ethernet OAM interface configuration.

Although you can configure multiple configuration templates, only one template can be associated with any single Ethernet OAM interface. You can override any commands defined within a template by explicitly configuring the same command (that is predefined by the template) in interface configuration mode.

To configure global Ethernet OAM interface options using a template, use the following command beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **template** *template-name*	Creates or selects a template and enters template configuration mode, where *template-name* is an up to 32-character string defining the name of the template.
Step 2	Router(config-template)# **ethernet oam link-monitor** *command* or Router(config-template)# **ethernet oam remote-failure** *command*	Specify one or more **ethernet oam** configuration commands. Repeat this step for the number of commands that you want to configure. For information about link monitoring commands, see the "Configuring Link Monitoring Options" section on page 12-31.
Step 3	Router(config-template)# **exit**	Exit template configuration mode and return to global configuration mode.

	Command	Purpose
Step 4	Router(config)# **interface** *type slot/subslot/port*	Specifies the Ethernet SPA interface, where • *type*—Specifies the type of Ethernet interface, such as **gigabitethernet** or **tengigabitethernet**. • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4. **Note** Ethernet OAM only can be defined on a main Gigabit Ethernet interface—not on subinterfaces.
Step 5	Router(config-if)# **source template** *template-name*	Attaches the template called *template-name* and applies the set of configuration commands defined by the named template to the specified interface.

Verifying Ethernet OAM Configuration

To verify the Ethernet OAM configuration, use the following commands in privileged EXEC configuration mode:

Command	Purpose
Router# **show ethernet oam discovery** [**interface** *type slot/subslot/port*]	Displays information about OAM functions negotiated during the OAM discovery phase of establishing an OAM session, where: • *type*—Specifies the type of Ethernet interface, such as **gigabitethernet** or **tengigabitethernet**. • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4.
Router# **show ethernet oam statistics** [**interface** *type slot/subslot/port*]	Displays statistics for information OAMPDUs and local and remote faults, where: • *type*—Specifies the type of Ethernet interface, such as **gigabitethernet** or **tengigabitethernet**. • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4

Command	Purpose
Router# **show ethernet oam status** [**interface** *type slot/subslot/port*]	Displays information about link monitoring configuration and status on the local OAM client, where: • *type*—Specifies the type of Ethernet interface, such as **gigabitethernet** or **tengigabitethernet**. • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4
Router# **show ethernet oam summary**	Displays information about the OAM session with the remote OAM client, where: • *type*—Specifies the type of Ethernet interface, such as **gigabitethernet** or **tengigabitethernet**. • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4

This section includes the following topics:

Verifying an OAM Session

To verify an OAM session, use the **show ethernet oam summary** command.

The following example shows that the local OAM client is established on the second Gigabit Ethernet SPA interface (1) located in subslot 1 of the SIP installed in chassis slot 6 of the Cisco 7600 series router (Gi6/1/1).

The local client interface is in session with a remote client with MAC address 0012.7fa6.a700 and organizationally unique identifier (OUI) 00000C, which is the OUI for Cisco Systems. The remote client is in active mode, and has established capabilities for link monitoring and remote loopback for the OAM session.

```
Router# show ethernet oam summary
Symbols:          * - Master Loopback State,  # - Slave Loopback State
Capability codes: L - Link Monitor,   R - Remote Loopback
                  U - Unidirection,   V - Variable Retrieval

  Local                          Remote
Interface       MAC Address    OUI    Mode    Capability

  Gi6/1/1       0012.7fa6.a700 00000C active   L R
```

Verifying OAM Discovery Status

To verify OAM Discovery status on the local client and remote peer, use the **show ethernet oam discovery** command as shown in the following example:

```
Router# show ethernet oam discovery interface gigabitethernet6/1/1

GigabitEthernet6/1/1
```

```
Local client
------------
  Administrative configurations:
    Mode:              active
    Unidirection:      not supported
    Link monitor:      supported (on)
    Remote loopback:   not supported
    MIB retrieval:     not supported
    Mtu size:          1500

  Operational status:
Port status:       operational
    Loopback status:   no loopback
    PDU permission:    any
    PDU revision:      1

Remote client
-------------
  MAC address: 0030.96fd.6bfa
  Vendor(oui): 0x00 0x00 0x0C (cisco)

  Administrative configurations:
    Mode:              active
    Unidirection:      not supported
    Link monitor:      supported
    Remote loopback:   not supported
    MIB retrieval:     not supported
    Mtu size:          1500
```

Verifying Information OAMPDU and Fault Statistics

To verify statistics for information OAMPDUs and local and remote faults, use the **show ethernet oam statistics** command as shown in the following example:

```
Router# show ethernet oam statistics interface gigabitethernet6/1/1

GigabitEthernet6/1/1
Counters:
---------
Information OAMPDU Tx                    : 588806
  Information OAMPDU Rx                    : 988
  Unique Event Notification OAMPDU Tx      : 0
  Unique Event Notification OAMPDU Rx      : 0
  Duplicate Event Notification OAMPDU TX   : 0
  Duplicate Event Notification OAMPDU RX   : 0
  Loopback Control OAMPDU Tx               : 1
  Loopback Control OAMPDU Rx               : 0
  Variable Request OAMPDU Tx               : 0
  Variable Request OAMPDU Rx               : 0
  Variable Response OAMPDU Tx              : 0
  Variable Response OAMPDU Rx              : 0
  Cisco OAMPDU Tx                          : 4
  Cisco OAMPDU Rx                          : 0
  Unsupported OAMPDU Tx                    : 0
  Unsupported OAMPDU Rx                    : 0
  Frames Lost due to OAM                   : 0

Local Faults:
-------------
  0 Link Fault records
  2 Dying Gasp records
    Total dying gasps       : 4
    Time stamp              : 00:30:39
```

```
    Total dying gasps       : 3
    Time stamp              : 00:32:39

  0 Critical Event records

Remote Faults:
--------------
  0 Link Fault records
  0 Dying Gasp records
  0 Critical Event records

Local event logs:
-----------------
  0 Errored Symbol Period records
  0 Errored Frame records
  0 Errored Frame Period records
  0 Errored Frame Second records

Remote event logs:
------------------
  0 Errored Symbol Period records
  0 Errored Frame records
  0 Errored Frame Period records
  0 Errored Frame Second records
```

Verifying Link Monitoring Configuration and Status

To verify link monitoring configuration and status on the local client, use the **show ethernet oam status** command. The highlighted "Status" field in the following example shows that link monitoring status is supported and enabled (on).

```
Router# show ethernet oam status interface gigabitethernet6/1/1

GigabitEthernet6/1/1
General
-------
  Mode:                    active
  PDU max rate:            10 packets per second
  PDU min rate:            1 packet per 1 second
  Link timeout:            5 seconds
  High threshold action: no action

Link Monitoring
---------------
  Status: supported (on)

  Symbol Period Error
    Window:                1 million symbols
    Low threshold:         1 error symbol(s)
    High threshold:        none

  Frame Error
    Window:                10 x 100 milliseconds
    Low threshold:         1 error frame(s)
    High threshold:        none
Frame Period Error
    Window:                1 x 100,000 frames
    Low threshold:         1 error frame(s)
    High threshold:        none

  Frame Seconds Error
    Window:                600 x 100 milliseconds
    Low threshold:         1 error second(s)
    High threshold:        none
```

Verifying Status of the Remote OAM Client

To verify the status of a remote OAM client, use the **show ethernet oam summary** and **show ethernet oam status** commands.

To verify the remote client mode and capabilities for the OAM session, use the **show ethernet oam summary** command and observe the values in the Mode and Capability fields. The following example shows that the local client (local interface Gi6/1/1) is connected to the remote client

```
Router# show ethernet oam summary
Symbols:          * - Master Loopback State,  # - Slave Loopback State
Capability codes: L - Link Monitor,  R - Remote Loopback
                  U - Unidirection,  V - Variable Retrieval

  Local                        Remote
Interface       MAC Address     OUI    Mode     Capability

  Gi6/1/1       0012.7fa6.a700 00000C active   L R
```

Configuring IP Subscriber Awareness over Ethernet

Container interfaces are used to apply hardware specific features like Security Access Control List (ACL) and Policy Based Routing (PBR) which then can be inherited to all IP session interfaces attached to a container interface.

To form the association between a container interface and an IP session interface/subinterface, use the **container** command under IP session interfaces/subinterfaces.

It is required to configure the VRF (not required in the case of global VRF) on the container and the subinterface in order to make an association between them using the **container** command.

	Command	Purpose
Step 1	Router(config)# **interface gigabitethernet** *slot/subslot/port.subinterface-number* **access**	Specifies the GigabitEthernet interface to configure, where: • *slot/subslot*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 12-4. • *port.subinterface-number*—Specifies a secondary interface (subinterface) number. • **access**—Indentifies the subscriber in the access-side network on subinterfaces.
Step 2	Router(config)# **ip vrf forwarding** *vrf-name*	Defines the VRF.
Step 3	Router(config-subif)# **container** *container number*	Defines the virtual interface and that would be allocated as the internal VLAN which would be shared by all the IP session interfaces which are tied with the container interface.
Step 4	Router(config-subif)# **encapsulation dot1q** *vlan-id*	Defines the encapsulation format as IEEE 802.1Q ("dot1q"), where *vlan-id* is the number of the VLAN (1–4095).

IP Subscriber Awareness over Ethernet Restrictions

There are restrictions being imposed because the internal VLAN is shared by multiple subinterfaces. The restrictions are as follows:

- IP Subscriber awareness over Ethernet is only supported on a Cisco 7600 SIP-400.
- Security ACL will not be supported on per IP subscriber interface basis. However, security ACL feature will be supported on a per group basis.
- Only single route-map policy can be applied on all subinterfaces that are sharing the Internal VLAN. If route-map is defined based on source IP address, then source IP address range should be easily definable and should not cause a configuration bloat.
- unicast Reverse Path Forwarding (uRPF) check can be done only on an internal VLAN level that is shared by subinterfaces, and not at subinterface level. Because of this restriction, a subscriber sharing the same internal VLAN may be able to spoof the IP address of some other subscribers.
- IPv4 multicast is not supported on IP session interfaces. IPv4 multicast does not have any functionality on a per-group basis, as replication is always required on a interface basis and not on a group basis.

There are also some configuration restrictions for link redundancy:

- There is no mechanism to synchronize the route installed by the DHCP to multiple routers; it will be difficult to use IP unnumbered' on and IP session interface. Instead, numbered IP addresses will be used on IP session interface and DHCP will assign IP addresses to the subscriber from the same subnet assigned to the IP session interface.
- It is required to configure the HSRP group for each IP session interface so the Cisco 7600 series router can scale to a 16K HSRP group.

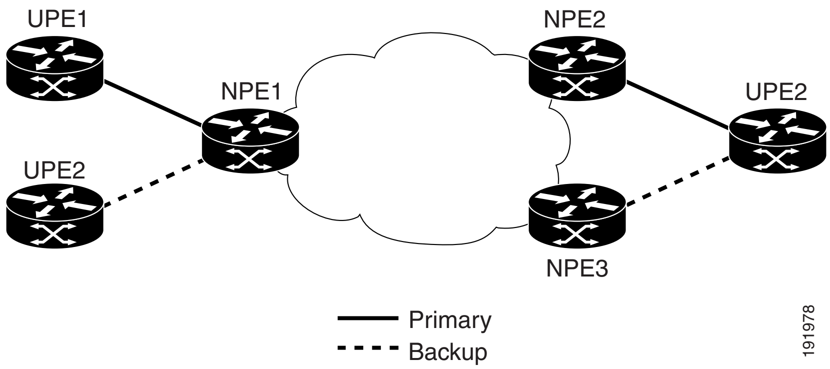
Configuring a Backup Interface for Flexible UNI

The Backup Interface for Flexible UNI feature allows you to configure redundant user-to-network interface (UNI) connections for Ethernet interfaces, which provides redundancy for dual-homed devices.

You can configure redundant (flexible) UNIs on a network provider-edge (N-PE) device in order to supply flexible services through redundant user provider-edge (U-PE) devices. The UNIs on the N-PEs are designated as primary and backup and have identical configurations. If the primary interface fails, the service is automatically transferred to the backup interface.

Figure 12-4 shows an example of how Flexible UNIs can be used when the Cisco 7600 series router is configured as a dual-homed N-PE (NPE1) and as a dual-homed U-PE (UPE2).

Figure 12-4 Backup Interface for Dual-Homed Devices



Note The configurations on the primary and backup interfaces must be identical.

The primary interface is the interface for which you configure a backup. During operation, the primary interface is active and the backup (secondary) interface operates in standby mode. If the primary interface goes down (due to loss of signal), the router begins using the backup interface.

While the primary interface is active (up) the backup interface is in standby mode. If the primary interface goes down, the backup interface transitions to the up state and the router begins using it in place of the primary. When the primary interface comes back up, the backup interface transitions back to standby mode. While in standby mode, the backup interface is effectively down and the router does not monitor its state or gather statistics for it.

This feature provides the following benefits:

- Supports the following Ethernet virtual circuit (EVC) features:
  - Frame matching: EVC with any supported encapsulation (Dot1q, default, untagged)
  - Frame rewrite: Any supported (ingress and egress with push, pop, and translate)
  - Frame forwarding: MultiPoint Bridging over Ethernet (MPB-E), xconnect, connect
  - Quality of Service (QoS) on EVC
- Supports Layer 3 (L3) termination and L3 VRF
- Supports several types of uplinks: MPLS, VPLS, and switchports

The Backup Interface for Flexible UNI feature makes use of these Ethernet components:

- Ethernet virtual circuit (EVC)—An association between two or more UNIs that identifies a point-to-point or point-to-multipoint path within the provider network. For more information about EVCs, see the description of “Flexible QinQ Mapping and Service Awareness” at the following URL:

  http://www.cisco.com/en/US/products/hw/routers/ps368/products_configuration_guide_chapter09186a00807f3f97.html#wp1433597
- Ethernet flow point (EFP)—The logical demarcation point of an EVC on an interface. An EVC that uses two or more UNIs requires an EFP on the associated ingress interface and egress interface of every device that the EVC passes through.

Configuration Guidelines

Observe these guidelines as you configure a backup interface for Flexible UNI on the router:

- Hardware and software support:
  - Supported on the Cisco 7600-ES20-2x10G and 7600-ES20-20x1G line cards.
  - Supported on the Cisco 7600 SIP-400 with Gigabit Ethernet SPAs. In an EVC configuration, version 2 SPAs are required. For IP termination, the SPAs can be version 1 or version 2.
  - Supported with the Route Switch Processor 720, Supervisor Engine 720, and Supervisor Engine 32.
  - Requires Cisco IOS Release 12.2SRB1 or later.
- You can use the same IP address on both the primary and secondary interfaces. This enables the interface to support L3 termination (single or double tagged).
- The configurations on the primary and backup interfaces must match. The router does not check that the configurations match; however, the feature does not work if the configurations are not the same.

  Note If the configuration includes the **xconnect** command, you must specify a different VCID on the primary and backup interfaces.

- The duplicate resources needed for the primary and secondary interfaces are taken from the total resources available on the router and thus affect available resources. For example, each **xconnect** consumes resources on both the primary and backup interfaces.
- Local switching (connect) between primary and backup interfaces uses twice the number of physical interfaces. This limitation is due to lack of support for local switching on EVCs on the same interface.
- Any features configured on the primary and backup interfaces (such as bridge-domain, xconnect, and connect) transition up or down as the interface itself transitions between states.
- Switchover time between primary and backup interfaces is best effort. The time it takes the backup interface to transition from standby to active mode depends on the link-state detection time and the amount of time needed for EVCs and their features to transition to the up state.
- Configuration changes and administrative actions made on the primary interface are automatically reflected on the backup interface.
- The router monitors and gathers statistics for the active interface only, not the backup. During normal operation, the primary interface is active; however, if the primary goes down, the backup becomes active and the router begins monitoring and gathering statistics for it.
- When the primary interface comes back up, the backup interface always transitions back to standby mode. Once the signal is restored on the primary interface, there is no way to prevent the interface from being restored as the primary.

Configuration Instructions

To configure a backup interface for a flexible UNI on an Ethernet port, perform the following steps:

Note You must apply the same configuration to both the primary and backup interfaces or the feature does not work. To configure EVC service instances on the interfaces, use the **service instance**, **encapsulation**, **rewrite**, **bridge-domain**, and **xconnect** commands. For information, see the following URLs:
http://www.cisco.com/en/US/products/hw/routers/ps368/products_configuration_guide_chapter09186a00807f3f97.html#wp1341419
http://www.cisco.com/en/US/products/hw/routers/ps368/products_configuration_guide_chapter09186a00807f3f97.html#wp1433480

	Command or Action	Purpose
Step 1	Router(config)# **interface** *type slot/subslot/port* Router(config)# **interface gigabitethernet3/0/0**	Selects the primary interface. This is the interface you are creating a backup interface for. For example, **interface gigabitEthernet 3/0/0** selects the interface for port 0 of the Gigabit Ethernet card installed in slot 3, subslot 0. • *type* specifies the interface type. Valid values are **gigabitethernet** or **tengigabitethernet**. • *slot/subslot/port* specifies the location of the interface.
Step 2	Router(config-if)# **backup interface** *type interface* Router(config-if)# **backup interface gigabitethernet4/0/1**	Selects the interface to serve as a backup interface.
Step 3	Router(config-if)# **backup delay** *enable-delay disable-delay* Router(config-if)# **backup delay 0 0**	(Optional) Specifies a time delay (in seconds) for enabling or disabling the backup interface. • *enable-delay* is the amount of time to wait after the primary interface goes down before bringing up the backup interface. • *disable-delay* is the amount of time to wait after the primary interface comes back up before restoring the backup interface to the standby (down) state **Note** For the backup interface for Flexible UNI feature, do not change the default delay period (0 0) or the feature may not work correctly.
Step 4	Router(config-if)# **backup load** *enable-percent disable-percent* Router(config-if)# **backup load 50 10**	(Optional) Specifies the thresholds of traffic load on the primary interface (as a percentage of the total capacity) at which to enable and disable the backup interface. • *enable-percent*—Activate the backup interface when the traffic load on the primary exceeds this percentage of its total capacity. • *disable-percent*—Deactivate the backup interface when the combined load of both primary and backup returns to this percentage of the primary's capacity. Applying the settings from the example to a primary interface with 10-MB capacity, the router enables the backup interface when traffic load on the primary exceeds 5 Mbytes (50%), and disables the backup when combined traffic on both interfaces falls below 1 MB (10%).
Step 5	Router(config-if)# **exit**	Exits interface configuration mode and returns to global configuration mode.

	Command or Action	Purpose
Step 6	Router(config)# **connect primary** *interface srv-inst interface srv-inst* Router(config)# **connect backup** *interface srv-inst interface srv-inst* Router(config)# **connect primary gi3/0/0 2 gi3/0/1 2** Router(config)# **connect backup gi4/0/0 2 gi4/0/1 2**	(Optional) Creates a local connection between a single service instance (*srv-inst*) on two different interfaces. The **connect primary** command creates a connection between primary interfaces, and **connect backup** creates a connection between backup interfaces. In the example, a local connection is configured between service instance 2 on primary interfaces (gi3/0/0 and gi3/0/1) and on backup interfaces (gi4/0/0 and gi4/0/1).
Step 7	Router(config)# **connect primary** *interface srv-inst1 interface srv-inst2* Router(config)# **connect backup** *interface srv-inst1 interface srv-inst2* Router(config)# connect primary gi3/0/0 2 gi3/0/0 3 Router(config)# connect backup gi4/0/0 2 gi4/0/0 3	(Optional) Enables local switching between different service instances (*srv-inst1* and *srv-inst2*) on the same port. Use the **connect primary** command to create a connection on a primary interface, and **connect backup** to create a connection on a backup interface. In the example, we are configuring local switching between service instances 2 and 3 on both the primary (gi3/0/0) and backup interfaces (gi4/0/0).
Step 8	Router(config-if)# **exit**	Exits interface configuration mode.

The following example shows a sample configuration in which:

- gi3/0/1 is the primary interface and gi4/0/1 is the backup interface.
- Each interface supports two service instances (2 and 4), and each service instance uses a different type of forwarding (**bridge-domain** and **xconnect**).
- The **xconnect** command for service instance 2 uses a different VCID on each interface.

```
int gi3/0/1
  backup interface gi4/0/1
  service instance 4 ethernet
    encapsulation dot1q 4
    rewrite ingress tag pop 1 symmetric
    bridge-domain 4
  service instance 2 ethernet
    encapsulation dot1q 2
    rewrite ingress tag pop 1 symmetric
    xconnect 10.0.0.0 2 encap mpls

int gi4/0/1

  service instance 4 ethernet
    encapsulation dot1q 4
    rewrite ingress tag pop 1 symmetric
    bridge-domain 4
  service instance 2 ethernet
    encapsulation dot1q 2
    rewrite ingress tag pop 1 symmetric
    xconnect 10.0.0.0 5 encap mpls
```

Verifying the Flexible UNI Backup Interface Configuration

This section lists the commands to display information about the primary and backup interfaces configured on the router. In the examples that follow, the primary interface is gi3/0/0 and the secondary (backup) interface is gi3/0/11.

- To display a list of backup interfaces, use the **show backup** command in privileged EXEC mode. Our sample output shows a single backup (secondary) interface:

```
NPE-11# show backup
Primary Interface     Secondary Interface    Status
-----------------     -------------------    ------
GigabitEthernet3/0/0  GigabitEthernet3/0/11  normal operation
```

- To display information about a primary or backup interface, use the **show interfaces** command in privileged EXEC mode. Issue the command on the interface for which you want to display information. The following examples show the output displayed when the command is issued on the primary (gi3/0/0) and backup (gi3/0/11) interfaces:

```
NPE-11# show int gi3/0/0
GigabitEthernet3/0/0 is up, line protocol is up (connected)
  Hardware is GigEther SPA, address is 0005.dc57.8800 (bia 0005.dc57.8800)
  Backup interface GigabitEthernet3/0/11, failure delay 0 sec, secondary disable delay
0 sec, kickin load not set, kickout load not set
[…]

NPE-11# show int gi3/0/11
GigabitEthernet3/0/11 is standby mode, line protocol is down (disabled)
```

If the primary interface goes down, the backup (secondary) interface is transitioned to the up state, as shown in the command output that follows. Notice how the command output changes if you reissue the **show backup** and **show interfaces** commands at this time: the status retrieved by the **show backup** status changes, the line protocol for gi3/0/0 is now down (notconnect), and the line protocol for gi3/0/11 is now up (connected).

```
NPE-11# !!! Link gi3/0/0 (active) goes down…
22:11:11: %LINK-DFC3-3-UPDOWN: Interface GigabitEthernet3/0/0, changed state to down
22:11:12: %LINK-DFC3-3-UPDOWN: Interface GigabitEthernet3/0/11, changed state to up
22:11:12: %LINEPROTO-DFC3-5-UPDOWN: Line protocol on Interface GigabitEthernet3/0/0,
changed state to down
22:11:13: %LINEPROTO-DFC3-5-UPDOWN: Line protocol on Interface GigabitEthernet3/0/11,
changed state to up

NPE-11# show backup
Primary Interface     Secondary Interface    Status
-----------------     -------------------    ------
GigabitEthernet3/0/0  GigabitEthernet3/0/11  backup mode

NPE-11# show int gi3/0/0
GigabitEthernet3/0/0 is down, line protocol is down (notconnect)
  Hardware is GigEther SPA, address is 0005.dc57.8800 (bia 0005.dc57.8800)
  Backup interface GigabitEthernet3/0/11, failure delay 0 sec, secondary disable delay
0 sec,

NPE-11# show int gi3/0/11
GigabitEthernet3/0/11 is up, line protocol is up (connected)
```

Flexible QinQ Mapping and Service Awareness on the 1-Port 10-Gigabit Ethernet SPA

The Flexible QinQ Mapping and Service Awareness on 1-Port 10-Gigabit Ethernet SPA feature allows service providers to offer triple-play services, residential Internet access from a digital subscriber line access multiplexer (DSLAM), and business Layer 2 and Layer 3 VPN by providing for termination of double-tagged dot1q frames onto a Layer 3 subinterface at the access node.

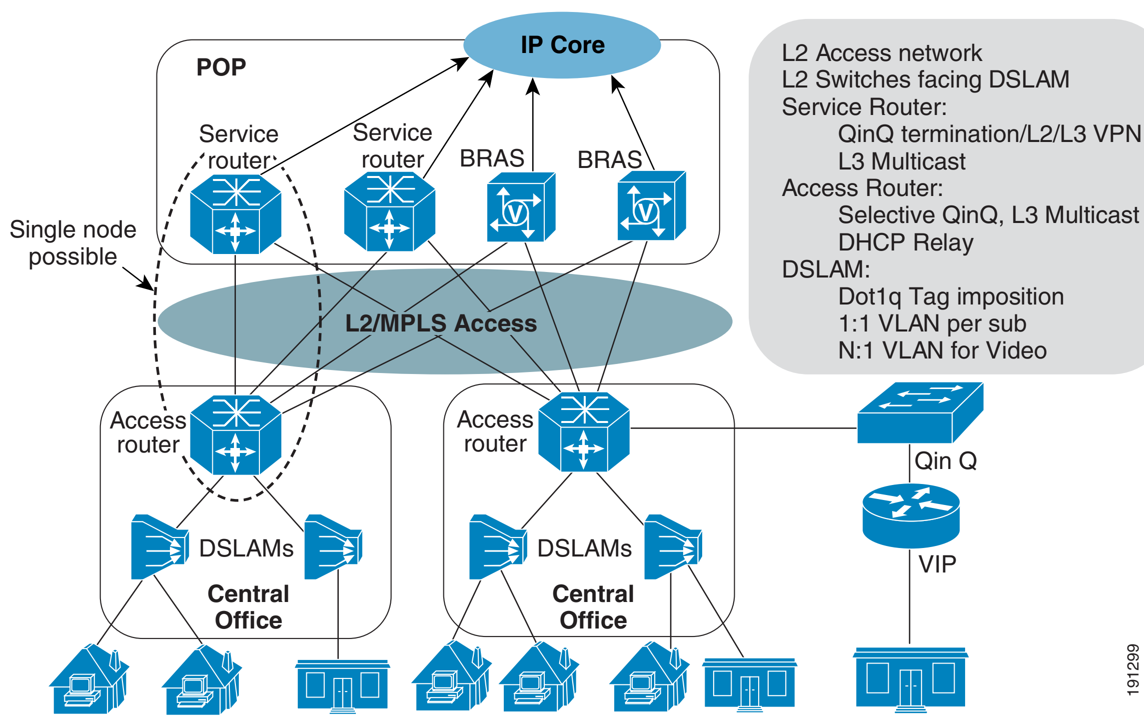
The access node connects to the DSLAM through the 1-Port 10-Gigabit Ethernet SPA. This provides a flexible way to identify the customer instance by its VLAN tags, and to map the customer instance to different services.

Flexible QinQ Mapping and Service Awareness on the1-Port 10-Gigabit Ethernet SPA is supported only through Ethernet Virtual Connection Services (EVCS) service instances.

EVCS uses the concepts of EVCs (Ethernet virtual circuits) and service instances. An EVC is an end-to-end representation of a single instance of a Layer 2 service being offered by a provider to a customer. It embodies the different parameters on which the service is being offered. A service instance is the instantiation of an EVC on a given port on a given router.

Figure 12-5 shows a typical metro architecture where the access switch facing the DSLAM provides VLAN translation (selective QinQ) and grooming funcitonality and where the serivce routers (SR) provide QinQ termination into a Layer 2 or Layer 3 service.

Figure 12-5 Typical Metro Architecture



Flexible QinQ Mapping and Service Awareness on the 1-Port 10-Gigabit Ethernet SPA provides the following functionality:

- VLAN connect with local significance (VLAN local switching)

   - Single tag Ethernet local switching where the received dot1q tag traffic from one port is cross-connected to another port by changing the tag. This is a 1-to-1 mapping service and there is no MAC learning involved.
   - Double tag Ethernet local switching where the received double tag traffic from one port is cross-connected to another port by changing both tags. The mapping to each double tag combination to the cross-connect is 1-to-1. There is no MAC learning involved.
- Selective QinQ (1-to-2 translation)
   - xconnect—Selective QinQ adds an outer tag to the received dot1q traffic and then tunnels it to the remote end with Layer 2 switching or EoMPLS.
   - Layer 2 switching—Selective QinQ adds an outer tag to the received dot1q traffic and then performs Layer 2 switching to allow switch virtual interface (SVI) based on the outer tag for configuring additional services.
- Double tag translation (2-to-2 translation) Layer 2 switching—Two received tagged frames are popped and two new tags are pushed.
- Double tag termination (2-to-1 tag translation)
   - Ethernet MultiPoint Bridging over Ethernet (MPBE)—The incoming double tag is uniquely mapped to a single dot1q tag that is then used to do MPBE
   - Double tag MPBE—The ingress line uses double tags in the ingress packet to look up the bridging VLAN. The double tags are popped and the egress line card adds new double tags and sends the packet out.
   - Double tag routing—Same as regular dot1q tag routing except that double tags are used to identify the hidden VLAN.
- Local VLAN significance—VLAN tags are significant only to the port.
- Scalable EoMPLS VC—Single tag packets are sent across the tunnel.
- QinQ policing and QoS
- Layer 2 protocol data unit (PDU) packet—If the Layer 2 PDUs are tagged, packets are forwarded transparently; if the Layer 2 PDUs are untagged, packets are treated per the physical port configuration.

Restrictions and Usage Guidelines

When configuring Flexible QinQ Mapping and Service Awareness on the 1-Port 10-Gigabit Ethernet SPA, follow these restrictions and usage guidelines:

- Service Scalability:
   - Service Instances: 16, 000
   - Input matching pairs: 8,000
   - Bridge-domains: 16, 000
   - Local switching: 32,000
   - Xconnect:16, 000
   - Subinterface: 2,000
- QoS Scalability:
   - Shaping: Parent queue is 2,000 and child queue is 16,000
   - Marking: Parent queue is 2,000 and child queue is 16,000

- Modular QoS CLI (MQC) actions supported include:
  - Shaping
  - Bandwidth
  - Two priority queues per policy
  - The **set cos** command, **set cos-inner** command, **set cos cos-inner** command, and **set cos-inner cos** command
  - WRED aggregate
  - Queue-limit

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface gigabitethernet** *slot*/*subslot*/*port*[.*subinterface-number*] or **interface tengigabitethernet** *slot*/*subslot*/*port*[.*subinterface-number*]
4. [**no**] **service instance** *id* {**Ethernet** *service-name*}
5. **encapsulation dot1q** *vlan-id*
6. **rewrite ingress tag** {**push** {**dot1q** *vlan-id* | **dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **pop** {**1** | **2**} | **translate** {**1-to-1** {**dot1q** *vlan-id* | **dot1ad** *vlan-id*}| **2-to-1 dot1q** *vlan-id* | **dot1ad** *vlan-id*}| **1-to-2** {dot1q *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **2-to-2** {**dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*}} [**symmetric**]

DETAILED STEPS

	Command	Purpose
Step 1	**enable** Router> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Router# **configure terminal**	Enters global configuration mode.
Step 3	**interface gigabitethernet** slot/subslot/port[.subinterface-number] or **interface tengigabitethernet** slot/subslot/port[.subinterface-number] Router(config)# **interface** *gigabitethernet 4/0/0*	Specifies the Gigabit Ethernet or the Ten Gigabit Ethernet interface to configure, where: • *slot/subslot/port*—Specifies the location of the interface. • *subinterface-number*—(Optional) Specifies a secondary interface (subinterface) number.

	Command	Purpose
Step 4	[**no**] **service instance** id {**Ethernet** [service-name} Router(config-if)# **service instance 101 ethernet**	Creates a service instance (an instantiation of an EVC) on an interface and sets the device into the config-if-srv submode.
Step 5	**encapsulation dot1q** *vlan-id* Router(config-if-srv)# **encapsulation dot1q 13**	Defines the matching criteria to be used in order to map ingress dot1q frames on an interface to the appropriate service instance.
Step 6	**rewrite ingress tag** {**push {dot1q** *vlan-id* | **dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **pop** {**1** | **2**} | **translate** {**1-to-1** {**dot1q** *vlan-id* | **dot1ad** *vlan-id*}| **2-to-1 dot1q** *vlan-id* | **dot1ad** *vlan-id*}| **1-to-2** {dot1q *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **2-to-2** {**dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*}} [**symmetric**] Router(config-if-srv)# **rewrite ingress tag push dot1q 20**	Specifies the tag manipulation that is to be performed on the frame ingress to the service instance.

Examples

Single Tag VLAN Connect

In this example, an incoming frame with a dot1q tag of 10 enters TenGigabitEthernet1/0/1. It is index directed to TenGigabitEthernet1/0/2 and exits with a dot1q tag of 11. No MAC learning is involved.

```
! DSLAM facing port
Router(config)# interface TenGigabitEthernet1/0/1
Router(config-if)# service instance 100 ethernet
Router(config-if-srv)# encapsulation dot1q 10
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
!L2 facing port
Router(config)# interface TenGigabitEthernet1/0/2
Router(config-if)# service instance 101 ethernet
Router(config-if-srv)# encapsulation dot1q 11
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
! connect service
Router# connect EVC1 TenGigabitEthernet1/0/1 100 TenGigabitEthernet1/0/2 101
```

Double Tag VLAN Connect

In this example, an incoming frame with an outer dot1q tag of 10 and inner tag of 20 enters TenGigabitEthernet1/0/1. It is index directed to TenGigabitEthernet1/0/2 and exits with an outer dot1q tag of 11 and inner tag 21. No MAC learning is involved.

```
! DSLAM facing port
Router(config)# interface TenGigabitEthernet1/0/1
Router(config-if)# service instance 100 ethernet
Router(config-if-srv)# encapsulation dot1q 10 second-dot1q 20
Router(config-if-srv)# rewrite ingress tag pop 2 symmetric
!L2 facing port
Router(config)# interface TenGigabitEthernet1/0/2
Router(config-if)# service instance 101 ethernet
Router(config-if-srv)# encapsulation dot1q 11 second-dot1q 21
Router(config-if-srv)# rewrite ingress tag pop 2 symmetric
! connect service
Router# connect EVC1 TenGigabitEthernet1/0/1 100 TenGigabitEthernet1/0/2 101
```

Selective QinQ with Connect

This configuration uses EoMPLS to perform packet forwarding. This is index directed.

```
! DSLAM facing port - single tag packet from link
Router(config)# interface TenGigabitEthernet1/0/1
Router(config-if)# service instance 100 ethernet
Router(config-if-srv)# encapsulation dot1q 10-20,30,50-60
!L2/QinQ facing port double tag packets
Router(config)# interface TenGigabitEthernet1/0/2
Router(config-if)# service instance 101 ethernet
Router(config-if-srv)# encapsulation dot1q 11 second-dot1q any
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
! connecting service instances
! QinQ outer dot1q is 11
Router# connect EVC1 TenGigabitEthernet1/0/1 100 TenGigabitEthernet1/0/2 101
```

Selective QinQ with Xconnect

This configuration uses EoMPLS to perform packet forwarding. This is not index directed.

```
! DSLAM facing port
Router(config)# interface TenGigabitEthernet1/0/1
Router(config-if)# service instance 100 ethernet
Router(config-if-srv)# encapsulation dot1q 10-20,30,50-60
Router(config-if-srv)# xconnect 2.2.2.2 999 pw-class vlan-xconnect
!
Router(config)# interface Loopback1
Router(config-if)# ip address 1.1.1.1 255.255.255.255
! MPLS core facing port
Router(config)# interface TenGigabitEthernet2/0/1
Router(config-if)# ip address 192.168.1.1 255.255.255.0
Router(config-if)# mpls ip
Router(config-if)# mpls label protocol ldp
! MPLS core facing port
Router(config)# interface TenGigabitEthernet2/0/1
Router(config-if)# ip address 192.169.1.1 255.255.255.0
Router(config-if)# mpls ip
Router(config-if)# mpls label protocol ldp
!
Router(config)# interface Loopback1
Router(config-if)# ip address 2.2.2.2 255.255.255.255

! CE facing EoMPLS configuration
Router(config)# interface TenGigabitEthernet1/0/2
Router(config-if)# service instance 1000
Router(config-if-srv)# encapsulation dot1q 1000 second-dot1q any
```

```
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
Router(config-if-srv)# xconnect 1.1.1.1 999 pw-class vlan-xconnect
```

Selective QinQ with Layer 2 Switching

This configuration uses Layer 2 Switching to perform packet forwarding. The forwarding mechanism is the same as MPB-E, only the rewrites for each service instance are different.

```
! DSLAM facing port, single tag incoming
Router(config)# interface TenGigabitEthernet1/0/1
Router(config-if)# service instance 100 ethernet
Router(config-if-srv)# encapsulation dot1q 10-20
Router(config-if-srv)# bridge-domain 11
! QinQ VLAN
Router(config)# interface VLAN11
!QinQ facing port
Router(config)# interface TenGigabitEthernet1/0/2
Router(config-if)# switchport
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk vlan allow 11
```

Double Tag Translation (2-to-2 Tag Translation)

In this case, double-tagged frames are received on ingress. Both tags are popped and two new tags are pushed. The packet is then Layer 2 switched to the bridge-domain VLAN.

```
! QinQ facing port
Router(config)# interface TenGigabitEthernet1/0/1
Router(config-if)# service instance 100 ethernet
Router(config-if-srv)# encapsulation dot1q 100 second-dot1q 10
Router(config-if-srv)# rewrite ingress tag translate 2-to-2 dot1q 200 second-dot1q 20
second-dot1q 10
Router(config-if-srv)# bridge-domain 200
! QinQ VLAN
Router(config)# interface VLAN200
!
Router(config)# interface TenGigabitEthernet1/0/2
Router(config-if)# service instance 101 ethernet
Router(config-if-srv)# encapsulation dot1q 200 second-dot1q 20
Router(config-if-srv)# bridge-domain 200
```

Double Tag Termination (2 to 1 Tag Translation)

This example falls under the Layer 2 switching case.

```
! Double tag traffic
Router(config)# interface TenGigabitEthernet1/0/1
Router(config-if)# service instance 100 ethernet
Router(config-if-srv)# encapsulation dot1q 200 second-dot1q 20
Router(config-if-srv)# rewrite ingress tag pop 2 symmetric
Router(config-if-srv)# bridge-domain 10
!
Router(config)# interface TenGigabitEthernet1/0/2
Router(config-if)# service instance 101 ethernet
Router(config-if-srv)# encapsulation dot1q 10
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
```

```
Router(config-if-srv)# bridge-domain 10
!
Router(config)# interface TenGigabitEthernet1/0/3
Router(config-if)# service instance 101 ethernet
Router(config-if-srv)# encapsulation dot1q 30
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
Router(config-if-srv)# bridge-domain 10
```

Verification

Use the following commands to verify operation.

Command	Purpose
Router# **show ethernet service evc** [**id** *evc-id* | **interface** *interface-id*] [**detail**]	Displays information pertaining to a specific EVC if an EVC ID is specified, or pertaining to all EVCs on an interface if an interface is specified. The **detailed** option provides additional information on the EVC.
Router# **show ethernet service instance** [**id** *instance-id* **interface** *interface-id* | **interface** *interface-id*] [**detail**]	Displays information about one or more service instances: If a service instance ID and interface are specified, only data pertaining to that particular service instance is displayed. If only an interface ID is specified, displays data for all service instances s on the given interface.
Router# **show ethernet service interface** [*interface-id*] [**detail**]	Displays information in the Port Data Block (PDB).
Router# **show mpls l2 vc detail**	Displays detailed information related to the virtual connection (VC).
Router# **show mpls forwarding**	Displays the contents of theMPLS Label Forwarding Information Base (LFIB). **Note** Output should have the label entry l2ckt.
Router# **show platform software efp-client**	Displays service instance details.

Configuring MultiPoint Bridging over Ethernet on the 1-Port 10-Gigabit Ethernet SPA

The MultiPoint Bridging over Ethernet (MPBE) on the 1-Port 10-Gigabit Ethernet SPA feature provides Ethernet LAN switching with MAC learning, local VLAN significance, and full QoS support. MPBE also provides Layer 2 switchport-like features without the full switchport implementation. MPBE is supported only through Ethernet Virtual Connection Services (EVCS) service instances.

EVCS uses the concepts of EVCs (Ethernet virtual circuits) and service instances. An EVC is an end-to-end representation of a single instance of a Layer 2 service being offered by a provider to a customer. It embodies the different parameters on which the service is being offered. A service instance is the instantiation of an EVC on a given port on a given router.

For MPBE, an EVC packet filtering capability prevents leaking of broadcast/multicast bridge-domain traffic packets from one service instance to another. Filtering occurs before and after the rewrite to ensure that the packet goes only to the intended service instance.

You can use MPBE to:

- Simultaneously configure Layer 2 and Layer 3 services such as Layer 2 VPN, Layer 3 VPN, and Layer 2 bridging on the same physical port.
- Define a broadcast domain in a system. Customer instances that are part of a broadcast domain can be in the same physical port or in different ports.
- Configure mutltiple service instances with different encapuslations and map them to a single bridge domain.
- Perform local switching between service instances under the same bridge domain.
- Span across different physical interfaces using service instances that are part of the same bridge domain.
- Use encapsulation VLANs as locally significant (physical port).
- Replicate flooded packets from the core to all service instances under the bridge domain.
- Configure a Layer 2 tunneling service or Layer 3 terminating service under the bridge domain VLAN.

MPBE accomplishes this by manipulating VLAN tags for each service instance and mapping the manipulated VLAN tags to Layer 2 or Layer 3 services. Possible VLAN tag manipulations include:

- Single tag termination
- Single tag tunneling
- Single tag translation
- Double tag termination
- Double tag tunneling
- Double tag translation
- Selective QinQ translation

Restrictions and Usage Guidelines

When configuring the MultiPoint Bridging over Ethernet on the 1-Port 10-Gigabit Ethernet SPA, follow these restrictions and usage guidelines:

- Each service instance is considered as a separate circuit under the bridge-domain.
- Encapsulation can be dot1q or QinQ packets.
- 60 MPB VCs are supported under one bridge-domain.
- Internet Group Management Protocol (IGMP) snooping is supported with MPB VCs.
- Split Horizon is supported with MPB VCs.
- Bridge protocol data unit (BDPU) packets are either tunneled or dropped.
- For ingress policing, only the drop action and the accept action for the **police** command are supported. Marking is not supported as part of the policing.
- Ingress shaping is not supported.
- For ingress marking, supports **match vlan** command, **match vlan-inner** command, **match cos** command, **match cos-inner** command, **set cos** command, and **set cos-inner** command.
- For egress marking, **set cos** command and **set cos-inner** command are supported; **match inner-cos** command and **match inner-vlan** command are not supported.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

interface gigabitethernet *slot/subslot/port*[.*subinterface-number*] or **interface tengigabitethernet** *slot/subslot/port*[.*subinterface-number*]

3. [**no**] **service instance** *id* {**Ethernet** [*service-name*}
4. **encapsulation dot1q** *vlan-id*
5. **rewrite ingress tag** {**push {dot1q** *vlan-id* | **dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **pop** {**1** | **2**} | **translate** {**1-to-1** {**dot1q** *vlan-id* | **dot1ad** *vlan-id*}| **2-to-1 dot1q** *vlan-id* | **dot1ad** *vlan-id*}| **1-to-2** {dot1q *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **2-to-2** {**dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*}} [**symmetric**]
6. [no] **bridge-domain** *bridge-id*

DETAILED STEPS

	Command	Purpose
Step 1	**enable** Router> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Router# **configure terminal**	Enters global configuration mode.
Step 3	**interface gigabitethernet** slot/subslot/port[.subinterface-number] or **interface tengigabitethernet** slot/subslot/port[.subinterface-number] Router(config)# **interface gigabitethernet4/0/0**	Specifies the Gigabit Ethernet or the Ten Gigabit Ethernet interface to configure, where: • *slot/subslot/port*—Specifies the location of the interface. • *subinterface-number*—(Optional) Specifies a secondary interface (subinterface) number.
Step 4	[**no**] **service instance** *id* {**Ethernet** *service-name*} Router(config-if)# **service instance 101 ethernet**	Creates a service instance (an instantiation of an EVC) on an interface and sets the device into the config-if-srv submode.
Step 5	**encapsulation dot1q** *vlan-id* Router(config-if-srv)# **encapsulation dot1q 10**	Defines the matching criteria to be used in order to map ingress dot1q frames on an interface to the appropriate service instance.

	Command	Purpose
Step 6	[**no**] **rewrite ingress tag** {**push {dot1q** *vlan-id* | **dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **pop** {**1** | **2**} | **translate** {**1-to-1** {**dot1q** *vlan-id* | **dot1ad** *vlan-id*}| **2-to-1 dot1q** *vlan-id* | **dot1ad** *vlan-id*}| **1-to-2** {dot1q vlan-id **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*} | **2-to-2** {**dot1q** *vlan-id* **second-dot1q** *vlan-id* | **dot1ad** *vlan-id* **dot1q** *vlan-id*}} [**symmetric**] Router(config-if-srv)# **rewrite ingress tag push dot1q 200**	This command specifies the tag manipulation that is to be performed on the frame ingress to the service instance. **Note** If this command is not configured, then the frame is left intact on ingress (the service instance is equivalent to a trunk port).
Step 7	[no] **bridge-domain** *bridge-id* Router(config-subif)# **bridge domain 12**	Binds the service instance to a bridge domain instance where *bridge-id* is the identifier for the bridge domain instance.

Examples

Single Tag Termination Example

In this example, the single tag termination indentifies customers based on a single VLAN tag and maps the single-VLAN tag to the bridge-domain.

```
Router(config)# interface TenGigabitEthernet1/2/0
Router(config-if)# service instance 10 ethernet
Router(config-if-srv)# encapsulation dot1q 10
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
Router(config-if-srv)# bridge domain 12
}
```

Single Tag Tunneling Example

In this single tag tunneling example, the incoming VLAN tag is not removed but continues with the packet.

```
Router(config)# interface TenGigabitEthernet1/2/0
Router(config-if)# service instance 10 ethernet
Router(config-if-srv)# encapsulation dot1q 10
Router(config-if-srv)# bridge-domain 200
```

Single Tag Translation Example

In this single-tag translation example, the incoming VLAN tag is removed and VLAN 200 is added to the packet.

```
Router(config)# interface TenGigabitEthernet3/0/0
Router(config-if)# service instance 10 ethernet
Router(config-if-srv)# encapsulation dot1q 10
Router(config-if-srv)# rewrite ingress tag translate 1-to-1 dot1q 200 symmetric
Router(config-if-srv)# bridge-domain 200
```

Double Tag Termination Configuration Example

In this double-tag termination example, the ingress receives double tags that indentify the bridge VLAN; the double tags are stripped (terminated) from the packet.

```
Router(config)# interface TenGigabitEthernet2/0/0
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# encapsulation dot1q 10 inner 20
Router(config-if-srv)# rewrite ingress tag pop 2 symmetric
Router(config-if-srv)# bridge-domain 200
Router(config-if)# service instance 2
Router(config-if-srv)# encapsulation dot1q 40 inner 30
Router(config-if-srv)# rewrite ingress tag pop 2 symmetric
Router(config-if-srv)# bridge-domain 200
```

Double-Tag Translation Configuration Example

In this example, double tagged frames are received on ingress. Both tags are popped and two new tags are pushed. The packet is then Layer 2-switched to the bridge-domain VLAN.

```
Router(config)# interface TenGigabitEthernet1/0/0
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# encapsulation dot1q 10 second-dot1q 20
Router(config-if-srv)# rewrite ingress tag translate 2-to-2 dot1q 40 second dot1q 30
symmetric
Router(config-if-srv)# bridge-domain 200
Router(config-if)# service instance 2 ethernet
Router(config-if-srv)# encapsulation dot1q 40 second-dot1q 30
Router(config-if-srv)# rewrite ingress tag translate 2-to-2 dot1q 10 second dot1q 20
symmetric
Router(config-if-srv)# bridge-domain 200
```

Selective QinQ Configuration Example

In this example, a range of VLANs is configured and plugged into a single MPB VC.

```
Router(config)# interface TenGigabitEthernet1/0/0
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# encapsulation dot1q 10-20
Router(config-if-srv)# bridge-domain 200

Router(config)# interface TenGigabitEthernet2/0/0
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# encapsulation dot1q 10-30
Router(config-if-srv)# bridge-domain 200
```

Untagged Traffic Configuration Example

In this example, untagged traffic is bridged to the bridge domain and forwarded to the switchport trunk.

```
Router(config)# interface GigabitEthernet2/0/1
Router(config-if)# no ip address
Router(config-if)# service instance 1 ethernet
Router(config-if-srv)# encapsulation untagged
Router(config-if-srv)# bridge-domain 11
Router(config)# interface TenGigabitEthernet1/0/0
Router(config-if)# switchport
Router(config-if)# switchport mode trunk
Router(config-if)# switchport allowed vlan 11
```

MPBE with Split Horizon Configuration Example

In this example, unknown unicast traffic is flooded on the bridge domain except for the interface from which the traffic originated.

```
Router(config)# interface GigabitEthernet2/0/0
Router(config-if)# no ip address
Router(config-if)# service instance 1000 ethernet
Router(config-if-srv)# encapsulation dot1q 100 second-dot1q 10-20
Router(config-if-srv)# bridge-domain 100 split-horizon
Router(config-if)# service instance 1001 ethernet
Router(config-if-srv)# encapsulation dot1q 101 second-dot1q 21-30
Router(config-if-srv)# bridge-domain 101 split-horizon
Router(config-if)# service instance 1010 ethernet
Router(config-if-srv)# encapsulation dot1q 100
Router(config-if-srv)# rewrite ingress tag symmetric translate 1-to-2 dot1q 10 
second-dot1q 100 symmetric
Router(config-if-srv)# bridge-domain 10 split-horizon
Router(config-if)# mls qos trust dscp
```

In this example, service instances are configured on Ethernet interfaces and terminated on the bridge domain.

```
Router(config)# interface GigabitEthernet2/0/0
Router(config-if)# service instance 100 ethernet
Router(config-if-srv)# encapsulation dot1q 1000
Router(config-if-srv)# bridge-domain 10

Router(config)# interface GigabitEthernet1/0/0
Router(config-if)# switchport
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan 10
```

In this example, VPLS is configured in the core with multiple bridge domains.

```
!
l2 vfi vpls10 manual
 vpn id 10
 neighbor 20.0.0.2 encapsulation mpls
!
l2 vfi vpls100 manual
 vpn id 100
 neighbor 20.0.0.2 encapsulation mpls
!
l2 vfi vpls11 manual
 vpn id 11
 neighbor 20.0.0.2 encapsulation mpls
!
interface Vlan100
 mtu 9216
 no ip address
 xconnect vfi vpls1
end
```

Verification

Use the following commands to verify operation.

.

Command	Purpose
Router# **show ethernet service evc** [**id** *evc-id* | **interface** *interface-id*] [**detail**]	Displays information pertaining to a specific EVC if an EVC ID is specified, or pertaining to all EVCs on an interface if an interface is specified. The **detail** option provides additional information on the EVC.
Router# **show ethernet service instance** [**id** *instance-id* **interface** *interface-id* | **interface** *interface-id*] [**detail**]	Displays information about one or more service instances: If a service instance ID and interface are specified, only data pertaining to that particular service instance is displayed. If only an interface ID is specified, displays data for all service instances on the given interface.
Router# **show ethernet service interface** [*interface-id*] [**detail**]	Displays information in the Port Data Block (PDB).
Router# **show mpls l2 vc detail**	Displays detailed information related to the virtual connection (VC).
Router# **show mpls forwarding** (Output should have the label entry l2ckt)	Displays the contents of the MPLS Label Forwarding Information Base (LFIB).
Router# **show platform software efp-client**	Displays service instance details.

Configuring QoS Features on Ethernet SPAs

The SIPs and SPAs support many QoS features using modular QoS CLI (MQC) configuration. For information about the QoS features supported by the Ethernet SPAs, see the "Configuring QoS Features on a SIP" section on page 4-61.

For Fast Ethernet SPAs and the 2-Port Gigabit Ethernet SPA, the following QoS behavior applies:

- In both the ingress and egress directions, all QoS features calculate packet size similarly to how packet size calculation is performed by the FlexWAN and Enhanced FlexWAN modules on the Cisco 7600 series router.
- Specifically, all features consider the IEEE 802.3 Layer 2 headers and the Layer 3 protocol payload. The CRC, interframe gap, and preamble are not included in the packet size calculations.

Note For Fast Ethernet SPAs, QoS cannot change the speed of an interface (for example, Fast Ethernet SPAs cannot change QoS settings whenever an interface speed is changed between 100M to 10M). When the speed is changed, the user must also adjust the QoS setting accordingly.

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), use the following command in privileged EXEC configuration mode:

Command	Purpose
Router# **copy running-config startup-config**	Writes the new configuration to NVRAM.

For information about managing your system image and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications that correspond to your Cisco IOS software release.

Shutting Down and Restarting an Interface on a SPA

You can shut down and restart any of the interface ports on a SPA independently of each other. Shutting down an interface stops traffic and enters the interface into an "administratively down" state.

There are no restrictions for online insertion and removal (OIR) on Fast Ethernet or Gigabit Ethernet SPAs. Fast Ethernet and Gigabit Ethernet SPAs can be removed from a SIP at any time. SIPs populated with any type of SPAs can be removed from the router at any time.

If you are preparing for an OIR of a SPA, it is not necessary to independently shut down each of the interfaces prior to deactivation of the SPA. The **hw-module subslot** [**x/y**] **reload** command automatically stops traffic on the interfaces and deactivates them along with the SPA in preparation for OIR.

In similar fashion, you do not need to independently restart any interfaces on a SPA after OIR of a SPA or SIP.

To shut down an interface on a SPA, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **shutdown**	Disables an interface.

To restart an interface on a SPA, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **no shutdown**	Restarts a disabled interface.

Verifying the Interface Configuration

Besides using the **show running-configuration** command to display your router configuration settings, you can use the **show interfaces gigabitethernet** command to get detailed information on a per-port basis for your Gigabit Ethernet SPAs, and the **show interfaces fastethernet** command to get detailed information on a per-port basis for your Fast Ethernet SPAs.

The following example provides sample output for interface port 1 on the SPA located in the top subslot (0) of the SIP that is installed in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces gigabitethernet 2/0/1
GigabitEthernet2/0/1 is up, line protocol is up
  Hardware is GigEther SPA, address is 000a.f330.2e40 (bia 000a.f330.2e40)
  Internet address is 2.2.2.1/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
  Full-duplex, 1000Mb/s, link type is force-up, media type is SX
  output flow-control is on, input flow-control is on
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 03:18:49, output 03:18:44, output hang never
```

```
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     1703 packets input, 638959 bytes, 0 no buffer
     Received 23 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 1670 multicast, 0 pause input
     1715 packets output, 656528 bytes, 0 underruns
     0 output errors, 0 collisions, 4 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 pause output
     0 output buffer failures, 0 output buffers swapped out
```

Configuration Examples

This section includes the following configuration examples:

Basic Interface Configuration Example

The following example shows how to enter global configuration mode to specify the interface that you want to configure, configure an IP address for the interface, and save the configuration. This example configures interface port 1 on the SPA that is located in subslot 0 of the SIP, that is installed in slot 3 of the Cisco 7600 series router:

```
! Enter global configuration mode.
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address.
!
Router(config)# interface gigabitethernet 3/0/1
!
! Configure an IP address.
!
Router(config-if)# ip address 192.168.50.1 255.255.255.0
!
```

```
! Start the interface.
!
Router(config-if)# no shut
!
! Save the configuration to NVRAM.
!
Router(config-if)# exit
Router# copy running-config startup-config
```

MAC Address Configuration Example

The following example changes the default MAC address on the interface to 1111.2222.3333:

```
! Enter global configuration mode.
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address
!
Router(config)# interface gigabitethernet 3/0/1
!
! Modify the MAC address.
!
Router(config-if)# mac-address 1111.2222.3333
```

MAC Address Accounting Configuration Example

The following example enables MAC Address Accounting:

Enter global configuration mode.

```
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
```

Enable MAC address accounting

```
Router(config)# ip accounting mac-address {input | output}
Router(config-if)# ip accounting ?
access-violations Account for IP packets violating access lists on this interface
mac-address Account for MAC addresses seen on this interface
output-packets Account for IP packets output on this interface
precedence Count packets by IP precedence on this interface
<cr>
Router(config-if)# ip accounting mac
Router(config-if)# ip accounting mac-address ?
input Source MAC address on received packets
output Destination MAC address on transmitted packets
Router(config-if)# ip accounting mac-address ip
Router(config-if)# ip accounting mac-address input ?
<cr>
```

Specify MAC address accounting for traffic entering the interface.

```
!
Router(config-if)# ip accounting mac-address input
! Specify MAC address accounting for traffic leaving the interface.
!
Router(config-if)# ip accounting mac-address output
Router(config-if)# end
```

Verify the MAC Address on the interface.

```
!
Router# show interfaces GigabitEthernet 4/0/2 mac-accounting
GigabitEthernet4/0/2
Input (511 free)
000f.f7b0.5200(26 ): 124174 packets, 7450440 bytes, last: 1884ms ago
Total: 124174 packets, 7450440 bytes
Output (511 free)
000f.f7b0.5200(26 ): 135157 packets, 8109420 bytes, last: 1884ms ago
Total: 135157 packets, 8109420 bytes
```

HSRP Configuration Example

The following section provides a configuration example of Router A and Router B, each belonging to three VRRP groups:

Router A

Enter global configuration mode.

```
!
Router# configure terminal
!
```

Enter configuration commands, one per line. End with CNTL/Z.

```
!
Router# interface ethernet 1/0
ip address 10.1.0.2 255.0.0.0
Router# vrrp 1 priority 120
Router# vrrp 1 authentication cisco
Router# vrrp 1 timers advertise 3
Router# vrrp 1 timers learn
Router# vrrp 1 ip 10.1.0.10
Router# vrrp 5 priority 100
Router# vrrp 5 timers advertise 30
Router# vrrp 5 timers learn
Router# vrrp 5 ip 10.1.0.50
Router# vrrp 100 timers learn
Router# no vrrp 100 preempt
Router# vrrp 100 ip 10.1.0.100
no shutdown
```

Router B

Enter global configuration mode.

```
!
Router# configure terminal
!
```

Enter configuration commands, one per line. End with CNTL/Z.

```
!
Router# interface ethernet 1/0
ip address 10.1.0.1 255.0.0.0
Router# vrrp 1 priority 100
Router# vrrp 1 authentication cisco
Router# vrrp 1 timers advertise 3
Router# vrrp 1 timers learn
Router# vrrp 1 ip 10.1.0.10
Router# vrrp 5 priority 200
Router# vrrp 5 timers advertise 30
Router# vrrp 5 timers learn
```

```
Router# vrrp 5 ip 10.1.0.50
Router# vrrp 100 timers learn
Router# no vrrp 100 preempt
Router# vrrp 100 ip 10.1.0.100
Router# no shutdown
```

In this configuration, each group has the following properties:

- Group 1:
  - Virtual IP address is 10.1.0.10.
  - Router A will become the master for this group with priority 120.
  - Advertising interval is 3 seconds.
  - Preemption is enabled.
- Group 5:
  - Router B will become master for this group with priority 200.
  - Advertising interval is 30 seconds.
  - Preemption is enabled.
- Group 100:
  - –Router A will become master for this group first because it has a higher IP address (10.1.0.2).
  - –Advertising interval is the default 1 second.
  - –Preemption is disabled.

MTU Configuration Example

The following example sets the interface MTU to 9216 bytes.

Note The SPA automatically adds an additional 38 bytes to the configured interface MTU size.

Enter global configuration mode.

```
!
Router# configure terminal
!
```

Enter configuration commands, one per line. End with CNTL/Z.

```
!
```

Specify the interface address

```
!
Router(config)# interface gigabitethernet 3/0/1
!
```

Configure the interface MTU.

```
!
Router(config-if)# mtu 9216
```

VLAN Configuration Example

The following example creates subinterface number 268 on SPA interface port 2 (the third port), and configures the subinterface on the VLAN with ID number 268, using IEEE 802.1Q encapsulation:

> **Note** The SPA does not support ISL encapsulation.

Enter global configuration mode.

```
!
Router# configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
!
```

Specify the interface address

```
!
Router(config)# interface gigabitethernet 3/0/1.268
!
```

Configure dot1q encapsulation and specify the VLAN ID.

```
!
Router(config-subif)# encapsulation dot1q 268
```

AToM over GRE Configuration Example

The following example illustrates an AToM over GRE tunnel configuration between PE1 and PE2.

PE1:

```
interface GigabitEthernet4/3/0
 ip address 25.25.25.1 255.255.255.0
 negotiation auto
end

interface Tunnel10
 ip unnumbered Loopback1
 mpls label protocol ldp
 mpls ip
 tunnel source 12.12.12.12
 tunnel destination 6.6.6.6
end

interface Loopback1
 ip address 13.13.13.13 255.255.255.255
end

interface Loopback10
 ip address 12.12.12.12 255.255.255.255
end
ip route 2.2.2.2 255.255.255.255 Tunnel10
```

PE2:

```
interface GigabitEthernet2/3/0
```

```
 ip address 26.26.26.2 255.255.255.0
 negotiation auto
end

interface Tunnel10
 ip unnumbered Loopback1
 mpls ip
 tunnel source 6.6.6.6
 tunnel destination 12.12.12.12
end

interface Loopback1
 ip address 7.7.7.7 255.255.255.255
end

interface Loopback0
 ip address 6.6.6.6 255.255.255.255
end

ip route 3.3.3.3 255.255.255.255 Tunnel10
```

mVPNoGRE Configuration Examples

The following example shows the commands to configure the mVPNoGRE feature on a Cisco 7600 SIP-400 interface or subinterface; however, this example uses a Cisco 7600 SIP-400 interface that does *not* support subinterfaces:

Enter global configuration mode.

```
!
Router# configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
!
```

Specify the Gigabit Ethernet interface to configure.

```
!
Router(config)# interface gigabitethernet 2/0/0
```

Attach a GRE Tunnel to a Cisco 7600 SIP-400 subinterface.

```
!
Router(config-if)# tunnel-interface tu1
```

Define the IP traffic that should be tunneled.

```
!
Router(config-if-ti)# ip route 10.0.0.1 255.255.255.0
Router(config-if-ti)# exit
```

When the **tunnel-interface** command is configured on the Cisco 7600 SIP-400 interface or subinterface, ip pim sparse-mode and tag-switching ip are automatically added to the interface. A static route to IP address contained on the **ip route** command is internally created. The following example shows the output of a **show running interface** command after adding or configuring the **tunnel-interface** command; however, this example uses a Cisco 7600 SIP-400 interface that does *not* support subinterfaces:

```
Router# show running interface gigabitethernet 2/0/0
!
interface gigabitethernet2/0/0
 ip address 10.1.0.1 255.255.255.0
```

```
 ip pim sparse-mode
 no keepalive
 tunnel-interface Tunnel1
    ip route 10.11.0.1 255.255.255.0
    exit-tunnel-interface
 tag-switching ip
 clock source internal
end
```

Note You do not need to configure a static route (globally or on the tunnel) to the BGP neighbor on the Cisco 7600 series router. This is automatically done by the **ip route** command under the **tunnel-interface** command on the Cisco 7600 SIP-400 interface or subinterface.

The following example illustrates the tunnel interface configuration on the Cisco 7600 series router:

```
interface Tunnel0
 ip address 10.0.0.1 255.255.255.0
 ip pim sparse-dense-mode
 mpls ip
 tunnel source 22.22.22.22
 tunnel destination 44.44.44.44
```

EoMPLS Configuration Example

The following example shows the commands to configure software-based EoMPLS:

Enter global configuration mode.

```
!
Router# configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
Router# vlan 101
!
Router(config)# interface VLAN101
Router(config-if)# xconnect 7.7.7.7 73829 encapsulation MPLS
!
Router(config)# interface gigabitethernet 4/1/0.1
Router(config-subif)# encapsulation dot1Q 100
```

The following example shows the commands to configure Scalable EoMPLS (only for a Cisco 7600 SIP-400 Ethernet interface):

```
Router(config)# interface GigabitEthernet 1/2/1
Router(config-if)# no ip address
Router(config-if)# no cdp enable
!
Router(config-if)# interface GigabitEthernet 1/2/1.2
Router(config-subif)# encapsulation dot1Q 2
Router(config-subif)# xconnect 5.5.5.5 20002 encapsulation mpls
!
!
Router(config-if)# interface GigabitEthernet 1/2/1.4095
Router(config-subif)# encapsulation dot1Q 4095
Router(config-subif)# xconnect 5.5.5.5 24095 encapsulation mpls
```

The following example shows the commands to configure hardware EoMPLS (other ethernet interfaces):

```
Router(config)# interface GigabitEthernet 1/1
```

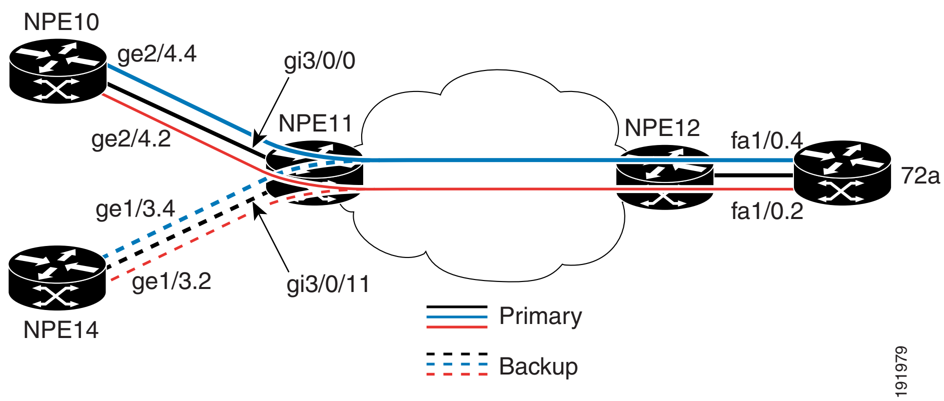
```
Router(config-if)# no ip address
Router(config-if)# no cdp enable
!
Router(config-subif)# interface GigabitEthernet 1/1.2
Router(config-subif)# encapsulation dot1Q 2
Router(config-subif)# xconnect 5.5.5.5 10002 encapsulation mpls
!
Router(config)# interface GigabitEthernet 1/1.3095
Router(config-subif)# encapsulation dot1Q 3095
Router(config-subif)# xconnect 5.5.5.5 13095 encapsulation mpls
!
```

Backup Interface for Flexible UNI Configuration Example

Figure 12-6 and the table that follows show a sample configuration that includes several EVCs (service instances), configured as follows:

- Service instance EVC4 is configured on primary and backup interfaces (links) that terminate in a bridge domain, with a VPLS uplink onto NPE12.
- Service instance EVC2 is configured as scalable Ethernet over MPLS, peering with an SVI VPLS on NPE12.

Figure 12-6 Backup Interface for Flexible UNI Configuration



NPE10 Configuration:

```
int ge2/4.4
  description npe10 to npe11 gi3/0/11 - backup - bridged
  encap dot1q 4
  ip address 100.4.1.33 255.255.255.0

int ge2/4.2
  description npe10 to npe11 gi3/0/11 - backup - xconnect
  encap dot1q 2
  ip address 100.2.1.33 255.255.255.0
```

U-PE2 Configuration:

```
int ge1/3.4
  description npe14 to npe11 gi3/0/0 - primary - bridged
  encap dot1q 4
  ip address 100.4.1.22 255.255.255.0

int ge1/3.2
  description npe14 to npe11 gi3/0/0 - primary - xconnect
  encap dot1q 2
  ip address 100.2.1.22 255.255.255.0
```

U-PE2 Configuration:

```
int fa1/0.4
  description 72a to npe12 - bridged
  encap dot1q 4
  ip address 100.4.1.12 255.255.255.0

int fa1/0.2
  description 72a to npe12 - xconnect
  encap dot1q 2
  ip address 100.2.1.12 255.255.255.0
```

<table>
<tr>
<td><pre>interface gigabitEthernet3/0/0
  backup interface gigabitEthernet3/0/11
  service instance 2 ethernet
    encapsulation dot1q 2
    rewrite ingress tag pop 1 symmetric
    xconnect 12.0.0.1 2 encapsulation mpls
  service instance 4 ethernet
    encapsulation dot1q 4
    rewrite ingress tag pop 1 symmetric
    bridge-domain 4

interface gigabitEthernet3/0/11
  service instance 2 ethernet
    encapsulation dot1q 2
    rewrite ingress tag pop 1 symmetric
    xconnect 12.0.0.1 21 encapsulation mpls
  service instance 4 ethernet
    encapsulation dot1q 4
    rewrite ingress tag pop 1 symmetric
    bridge-domain 4</pre></td>
<td><pre>interface GE-WAN 4/3
  description npe11 to npe12
  ip address 10.3.3.1 255.255.255.0
  mpls ip
l2 vfi vlan4 manual
  vpn id 4
  neighbor 12.0.0.1 4 encapsulation mpls
interface Vlan 4
  xconnect vfi vlan4</pre></td>
</tr>
<tr>
<td><pre>l2 vfi vlan4 manual
  vpn id 4
  neighbor 11.0.0.1 4 encap mpls
interface Vlan4
  description npe12 to npe11 xconnect
  xconnect vfi vlan4
l2 vfi vlan2 manual
  vpn id 2
  neighbor 11.0.0.1 2 encap mpls
  neighbor 11.0.0.1 21 encap mpls
Interface Vlan2
  xconnect vfi vlan2
interface GE-WAN 9/4
  description npe12 to npe11
  ip address 10.3.3.2 255.255.255.0
  mpls ip</pre></td>
<td><pre>interface fastEthernet 8/2
  description npe12 to 72a
  switchport
  switchport trunk encap dot1q
  switchport mode trunk
  switchport trunk allowed vlan 2-4</pre></td>
</tr>
</table>

Changing the Speed of a Fast Ethernet SPA Configuration Example

The following example shows the commands to change the speed of a Fast Ethernet SPA:

Note In order to change the speed of a Fast Ethernet SPA, autonegotiation must be disabled.

```
Router# show run interface fastethernet 5/0/1
Building configuration...
Current configuration : 86 bytes
!
Disable Autonegotiation
!
interface FastEthernet5/0/1
ip address 10.1.0.2 255.255.0.0
negotiation auto
end
Router# configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
Router(config)# interface fastethernet 5/0/1
Router(config-if)# no negotiation auto
Router(config-if)# speed 10
Router(config-if)#
Router(config-if)# end
Router# show run interface fastethernet 5/01
Building configuration...
Current configuration : 112 bytes
!
interface FastEthernet 5/0/1
ip address 10.1.0.2 255.255.0.0
speed 10
duplex full
no negotiation auto
end
Router# show interface fastethernet 5/0/1
FastEthernet5/0/1 is up, line protocol is up
Hardware is FastEthernet SPA, address is 000a.8b3e.cc00 (bia 000a.8b3e.cc00)
Internet address is 10.1.0.2/16
MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive not supported
Full-duplex, 10Mb/s
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:04, output 00:00:04, output hang never
Last clearing of "show interface" counters 1d00h
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
1608 packets input, 547102 bytes, 0 no buffer
Received 1 broadcasts (0 IP multicasts)
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog
0 input packets with dribble condition detected
1606 packets output, 548403 bytes, 0 underruns
Router# configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
Router(config)# interface fastethernet 5/0/1
Router(config-if)# speed 100
Router(config-if)# end
Router#
*Apr 25 21:10:36: %SYS-5-CONFIG_I: Configured from console by console
Router# show interface fastethernet 5/0/1
FastEthernet5/0/1 is down, line protocol is down
Hardware is FastEthernet SPA, address is 000a.8b3e.cc00 (bia 000a.8b3e.cc00)
Internet address is 10.1.0.2/16
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive not supported
Full-duplex, 100Mb/s
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:23, output 00:00:22, output hang never
Last clearing of "show interface" counters 1d00h
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
```

```
5 minute output rate 0 bits/sec, 0 packets/sec
1608 packets input, 547102 bytes, 0 no buffer
Received 1 broadcasts (0 IP multicasts)
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
```

Ethernet OAM Configuration Example

The following Ethernet OAM example shows configuration of Ethernet OAM options using a template, and overriding that configuration with direct configuration at an interface. In this example, the network supports a Gigabit Ethernet interface between the customer edge device and provider edge device:

Configure a global OAM template for both PE and CE configuration.

```
!
Router(config)# template oam
Router(config-template)# ethernet oam link-monitor symbol-period threshold low 10
Router(config-template)# ethernet oam link-monitor symbol-period threshold high 100
Router(config-template)# ethernet oam link-monitor frame window 100
Router(config-template)# ethernet oam link-monitor frame threshold low 10
Router(config-template)# ethernet oam link-monitor frame threshold high 100
Router(config-template)# ethernet oam link-monitor frame-period window 100
Router(config-template)# ethernet oam link-monitor frame-period threshold low 10
Router(config-template)# ethernet oam link-monitor frame-period threshold high 100
Router(config-template)# ethernet oam link-monitor frame-seconds window 1000
Router(config-template)# ethernet oam link-monitor frame-seconds threshold low 10
Router(config-template)# ethernet oam link-monitor frame-seconds threshold high 100
Router(config-template)# ethernet oam link-monitor receive-crc window 100
Router(config-template)# ethernet oam link-monitor receive-crc threshold high 100
Router(config-template)# ethernet oam link-monitor transmit-crc window 100
Router(config-template)# ethernet oam link-monitor transmit-crc threshold high 100
Router(config-template)# ethernet oam remote-failure dying-gasp action
error-disable-interface
Router(config-template)# exit
!
! Enable Ethernet OAM on the CE interface
!
Router(config)# interface gigabitethernet 4/1/1
Router(config-if)# ethernet oam
!
! Apply the global OAM template named "oam" to the interface.
!
Router(config-if)# source template oam
!
! Configure any interface-specific link monitoring commands to
! override the template configuration. The following example disables the high threshold
! link monitoring for receive CRC errors.
!
Router(config-if)# ethernet oam link-monitor receive-crc threshold high none
!
! Enable Ethernet OAM on the PE interface
!
Router(config)# interface gigabitethernet 8/1/1
Router(config-if)# ethernet oam
!
! Apply the global OAM template named "oam" to the interface.
!
Router(config-if)# source template oam
```

CHAPTER 13

Troubleshooting the Fast Ethernet and Gigabit Ethernet SPAs

This chapter describes techniques that you can use to troubleshoot the operation of your Fast Ethernet or Gigabit Ethernet SPAs.

It includes the following sections:

The first section provides information about basic interface troubleshooting. If you are having a problem with your SPA, use the steps in the "Performing Basic Interface Troubleshooting" section to begin your investigation of a possible interface configuration problem.

To perform more advanced troubleshooting, see the other sections in this chapter.

General Troubleshooting Information

This section describes general information for troubleshooting SIPs and SPAs. It includes the following sections:

Using debug Commands

Along with the other **debug** commands supported on the Cisco 7600 series router, you can obtain specific debug information for SPAs on the Cisco 7600 series router using the **debug hw-module subslot** privileged EXEC command.

The **debug hw-module subslot** command is intended for use by Cisco Systems technical support personnel.

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

For information about other **debug** commands supported on the Cisco 7600 series router, refer to the *Cisco IOS Debug Command Reference* and any related feature documents for the applicable Cisco IOS release.

Using show Commands

There are several **show** commands that you can use to monitor and troubleshoot the SIPs and SPAs on the Cisco 7600 series router. This chapter describes using the **show interfaces** command to perform troubleshooting of your SPA.

For more information about **show** commands to verify and monitor SIPs and SPAs, see the following chapters of this guide:

- Chapter 12, "Configuring the Fast Ethernet and Gigabit Ethernet SPAs"

Performing Basic Interface Troubleshooting

You can perform most of the basic interface troubleshooting using the **show interfaces fastethernet**, **show interfaces gigabitethernet**, or **show interfaces tengigabitethernet** command and examining several areas of the output to determine how the interface is operating.

The following example shows output from both the **show interfaces fastethernet, show interfaces gigabitethernet** and **show interfaces tengigabitethernet** commands with some of the significant areas of the output to observe shown in bold:

```
Router# show interfaces fastethernet 3/2/3
FastEthernet3/2/3 is up, line protocol is up
Hardware is FastEthernet SPA, address is 000e.d623.e840 (bia 000e.d623.e840)
Internet address is 33.1.0.2/16
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
reliability 255/255, txload 59/255, rxload 83/255
Encapsulation ARPA, loopback not set
Keepalive not supported
Full-duplex, 100Mb/sARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:11, output 00:00:08, output hang never
Last clearing of "show interface" counters 3d00h
Input queue: 0/75/626373350/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 32658000 bits/sec, 68032 packets/sec
5 minute output rate 23333000 bits/sec, 48614 packets/sec
    17792456686 packets input, 1067548381456 bytes, 0 no buffer
    Received 0 broadcasts (0 IP multicasts)
    0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 130043940 overrun, 0 ignored
    0 watchdog
    0 input packets with dribble condition detected
    12719598014 packets output, 763177809958 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
```

```
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out

Router# show interfaces gigabitethernet 2/0/1
GigabitEthernet2/0/1 is down, line protocol is down
  Hardware is GigEther SPA, address is 000a.f330.2e40 (bia 000a.f330.2e40)
  Internet address is 2.2.2.1/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
  Full-duplex, 1000Mb/s, link type is force-up, media type is SX
  output flow-control is on, input flow-control is on
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 03:18:49, output 03:18:44, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     1703 packets input, 638959 bytes, 0 no buffer
     Received 23 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 1670 multicast, 0 pause input
     1715 packets output, 656528 bytes, 0 underruns
     0 output errors, 0 collisions, 4 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 pause output
     0 output buffer failures, 0 output buffers swapped out

Router# show interfaces tengigabitethernet 7/0/0
TenGigabitEthernet7/0/0 is up, line protocol is up (connected)
  Hardware is TenGigEther SPA, address is 0000.0c00.0102 (bia 000f.342f.c340)
  Internet address is 15.1.1.2/24
  MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
  Full-duplex, 10Gb/s
  input flow-control is on, output flow-control is on
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 00:00:10, output hang never
  Last clearing of "show interface" counters 20:24:30
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  L2 Switched: ucast: 0 pkt, 0 bytes - mcast: 0 pkt, 0 bytes
  L3 in Switched: ucast: 0 pkt, 0 bytes - mcast: 0 pkt, 0 bytes mcast
  L3 out Switched: ucast: 0 pkt, 0 bytes mcast: 0 pkt, 0 bytes
     237450882 packets input, 15340005588 bytes, 0 no buffer
     Received 25 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 0 multicast, 0 pause input
     0 input packets with dribble condition detected
     1676 packets output, 198290 bytes, 0 underruns
     0 output errors, 0 collisions, 4 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 PAUSE output
```

```
0 output buffer failures, 0 output buffers swapped out
```

To verify that your interface is operating properly, complete the steps in Table 13-1:

Table 13-1 Basic Interface Troubleshooting Steps

	Action	Example

Step 1	From global configuration mode, enter the **show interfaces fastethernet**, **show interfaces gigabitethernet** or **show interfaces tengigabitethernet** command.	`Router# show interfaces fastethernet 3/2/3`
		
		`Router# show interfaces gigabitethernet 2/0/1`
		
		`Router# show interfaces tengigabitethernet 7/0/0`

Step 2	Verify that the interface is up.	`Router# show interfaces fastethernet 3/2/3`
		`FastEthernet3/2/3 is up, line protocol is up`
		
		`Router# show interfaces gigabitethernet 2/0/1`
		`GigabitEthernet2/0/1 is up, line protocol is up`
		
		`Router# show interfaces tengigabitethernet 7/0/0`
		`TenGigabitEthernet7/0/0 is up, line protocol is up`
		`(connected)`

Step 3	Verify that the line protocol is up.	`Router# show interfaces fastethernet 3/2/3`
		`FastEthernet3/2/3 is up, line protocol is up`
		
		`Router# show interfaces gigabitethernet 2/0/1`
		`GigabitEthernet2/0/1 is up, line protocol is up`
		
		`Router# show interfaces tengigabitethernet 7/0/0`
		`TenGigabitEthernet7/0/0 is up, line protocol is up`
		`(connected)`

Step 4	Verify that the interface duplex mode matches the remote interface configuration.	The following example shows that the local interface is currently operating in full-duplex mode:
		`Router# show interfaces fastethernet 3/2/3`
		`[text omitted]`
		
		`Keepalive not supported`
		`Full-duplex, 100Mb/sARP type: ARPA, ARP Timeout 04:00:00`
		
		`Router# show interfaces gigabitethernet 2/0/1`
		`[text omitted]`
		
		`Keepalive not supported`
		`Full-duplex, 1000Mb/s, link type is force-up, media type`
		`is SX`
		
		`Router# show interfaces tengigabitethernet 7/0/0`
		`[text omitted]`
		
		`Keepalive not supported`
		`Full-duplex, 10Gb/s`

Table 13-1 Basic Interface Troubleshooting Steps (continued)

	Action	Example

Step 5	Verify that the interface speed matches the speed on the remote interface.	The following example shows that the local interface is currently operating at 100 Mbps (Fast Ethernet and Gigabit Ethernet) or 10 Gbps (Ten Gigabit Ethernet):
		`Router# show interfaces fastethernet 3/2/3`
		`[text omitted]`
		
		`Keepalive not supported`
		`Full-duplex, 100Mb/sARP type: ARPA, ARP Timeout 04:00:00`
		
		`Router# show interfaces gigabitethernet 2/0/1`
		`[text omitted]`
		
		`Keepalive not supported`
		`Full-duplex, 1000Mb/s, link type is force-up, media type is SX`
		
		`Router# show interfaces tengigabitethernet 7/0/0`
		`[text omitted]`
		`Full-duplex, 10Gb/s`

Step 6	Observe the output hang status on the interface.	`ARP type: ARPA, ARP Timeout 04:00:00` `Last input 03:18:49, output 03:18:44, output hang never`
Step 7	Observe the CRC counter.	`0 input errors, 0 CRC, 0 frame, 130043940 overrun, 0 ignored`
Step 8	Observe the late collision counter.	`0 output errors, 0 collisions, 4 interface resets` `0 babbles, 0 late collision, 0 deferred`
Step 9	Observe the carrier signal counters.	`0 lost carrier, 0 no carrier, 0 pause output` `0 output buffer failures, 0 output buffers swapped out`

For more information about the verification steps and possible responses to correct detected problems, see the following sections:

- Verifying the Interface Is Up, page 13-5
- Verifying the Line Protocol Is Up, page 13-6
- Verifying Output Hang Status, page 13-6
- Verifying the CRC Counter, page 13-6
- Verifying Late Collisions, page 13-6
- Verifying the Carrier Signal, page 13-7

Verifying the Interface Is Up

In the output from the **show interfaces fastethernet**, **show interfaces gigabitethernet** or **show interfaces tengigabitethernet** command, verify that the interface is up. If the interface is down, perform the following corrective actions:

- If the interface is *administratively down*, use the **no shutdown** interface configuration command to enable the interface.
- Be sure that the cable is fully connected.
- Verify that the cable is not bent or damaged. If the cable is bent or damaged, the signal will be degraded.

- Verify that a hardware failure has not occurred. Observe the LEDs to confirm the failure. See the other troubleshooting sections of this chapter, and refer to the *Cisco 7600 Series Router SIP, SSC, and SPA Hardware Installation Guide*. If the hardware has failed, replace the SPA as necessary.

Verifying the Line Protocol Is Up

In the output from the **show interfaces fastethernet**, **show interfaces gigabitethernet** or **show interfaces tengigabitethernet** command, verify that the line protocol is up. If the line protocol is down, the line protocol software processes have determined that the line is unusable.

Perform the following corrective actions:

- Replace the cable.
- Check the local and remote interface for misconfiguration.
- Verify that a hardware failure has not occurred. Observe the LEDs to confirm the failure. See the other troubleshooting sections of this chapter, and refer to the Cisco 7600 Series Router SIP, SSC, and SPA Hardware Installation Guide. If the hardware has failed, replace the SPA as necessary.

Verifying Output Hang Status

In the output from the **show interfaces fastethernet**, **show interfaces gigabitethernet** or **show interfaces tengigabitethernet** command, observe the value of the output hang field.

The output hang provides the number of hours, minutes, and seconds since the last reset caused by a lengthy transmission. When the number of hours in the field exceeds 24 hours, the number of days and hours is shown. If the field overflows, asterisks are printed. The field shows a value of *never* if no output hangs have occurred.

Verifying the CRC Counter

In the output from the **show interfaces fastethernet**, **show interfaces gigabitethernet** or **show interfaces tengigabitethernet** command, observe the value of the CRC counter. Excessive noise will cause high CRC errors accompanied by a low number of collisions.

Perform the following corrective actions if you encounter high CRC errors:

- Check the cables for damage.
- Verify that the correct cables are being used for the SPA interface.

Verifying Late Collisions

In the output from the **show interfaces fastethernet**, **show interfaces gigabitethernet** or **show interfaces tengigabitethernet** command, observe the value of the late collision counter.

Perform the following corrective actions if you encounter late collisions on the interface:

- Verify that the duplex mode on the local and remote interface match. Late collisions occur when there is a duplex mode mismatch.
- Verify the length of the Ethernet cables. Late collisions result from cables that are too long.

Verifying the Carrier Signal

In the output from the **show interfaces fastethernet**, **show interfaces gigabitethernet** or **show interfaces tengigabitethernet** command, observe the value of the carrier signal counters. The lost carrier counter shows the number of times that the carrier was lost during transmission. The no carrier counter shows the number of times that the carrier was not present during transmission.

Carrier signal resets can occur when an interface is in loopback mode or shut down.

Perform the following corrective actions if you observe the carrier signal counter incrementing outside of these conditions:

- Check the interface for a malfunction.
- Check for a cable problem.

Understanding SPA Automatic Recovery

When Fast Ethernet or Gigabit Ethernet SPAs encounter thresholds for certain types of errors and identify a fatal error, the SPAs initiate an automatic recovery process.

You do not need to take any action unless the error counters reach a certain threshold, and multiple attempts for automatic recovery by the SPA fail.

The Gigabit Ethernet SPAs might perform automatic recovery for the following types of errors:

- SPI4 TX/RX out of frame
- SPI4 TX train valid
- SPI4 TX DIP4
- SPI4 RX DIP2

When Automatic Recovery Occurs

If the SPI4 errors occur more than 25 times within 10 milliseconds, the SPA automatically deactivates and reactivates itself. Error messages are logged on the console indicating the source of the error and the status of the recovery.

If Automatic Recovery Fails

If the SPA attempts automatic recovery more than five times in an hour, then the SPA deactivates itself and remains deactivated.

To troubleshoot automatic recovery failure for a SPA, perform the following steps:

Step 1 Use the **show hw-module subslot** *slot*/*subslot* **oir** command to verify the status of the SPA. The status is shown as “failed” if the SPA has been powered off due to five consecutive failures.

Step 2 If you verify that automatic recovery has failed, perform OIR of the SPA. For information about performing OIR, see the “Preparing for Online Insertion and Removal of a SPA” section on page 13-10.

Step 3 If reseating the SPA after OIR does not resolve the problem, replace the SPA hardware.

Configuring the Interface for Internal and External Loopback

Loopback support is useful for testing the interface without connectivity to the network, or for diagnosing equipment malfunctions between the interface and a device. The Fast Ethernet and Gigabit Ethernet SPAs support both an internal and an external loopback mode. The external loopback mode requires the use of a loopback cable and implements a loopback through the transceiver on the SPA.

You can also configure an internal loopback without the use of a loopback cable that implements a loopback at the PHY device internally on a Fast Ethernet or Gigabit Ethernet interface port, or at the MAC device internally on a Fast Ethernet or Gigabit Ethernet interface port. By default, loopback is disabled.

Configuring the Interface for Internal Loopback

Different Fast Ethernet and Gigabit Ethernet interfaces use different **loopback** commands.

To enable internal loopback at the PHY device for an interface on a SPA, use one of the following commands beginning in interface configuration mode:

Command or Action	Purpose
Router(config-if)# **loopback**	Enables an interface for internal loopback on the Gigabit Ethernet SPA.
Router(config-if)# **loopback internal**	Enables an interface for internal loopback on the Gigabit Ethernet SPA.
Router(config-if)# **loopback mac**	Enables an interface for internal loopback at the MAC controller level on the Fast Ethernet SPA.
Router(config-if)# **loopback driver**	Enables an interface for internal loopback at the transceiver level on the Fast Ethernet SPA.

Configuring the Interface for External Loopback

Before beginning external loopback testing, remember that the external loopback mode requires the use of a loopback cable.

External loopback cannot be configured on Fast Ethernet SPAs. To enable external loopback on Gigabit Ethernet SPAs, use the following commands beginning in interface configuration mode:

Command	Purpose
Router(config-if)# **loopback external**	Enables an interface for external loopback on the Gigabit Ethernet SPA.

Verifying Loopback Status

To verify whether loopback is enabled on an interface port on a SPA, use the **show interfaces fastethernet**, **show interfaces gigabitethernet** or **show interfaces tengigabitethernet** in privileged EXEC command and observe the value shown in the “loopback” field.

The following example shows that loopback is disabled for interface port 3 on the Fast Ethernet SPA installed in subslot 2 of the SIP that is located in slot 3 of the Cisco 7600 series router:

```
Router# show interfaces fastethernet 3/2/3
FastEthernet3/2/3 is up, line protocol is up
Hardware is FastEthernet SPA, address is 000e.d623.e840 (bia 000e.d623.e840)
Internet address is 33.1.0.2/16
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
reliability 255/255, txload 59/255, rxload 83/255
Encapsulation ARPA, loopback not set
```

The following example shows that loopback is disabled for interface port 0 (the first port) on the Gigabit Ethernet SPA installed in the top (0) subslot of the SIP that is located in slot 3 of the Cisco 7600 series router:

```
Router# show interfaces gigabitethernet 3/0/0
GigabitEthernet3/0/0 is up, line protocol is up
  Hardware is GigMac 1 Port 10 GigabitEthernet, address is 0008.7db3.8dfe (bia )
  Internet address is 10.0.0.2/24
  MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set
```

The following example shows that loopback is disabled for interface port 0 (the first port) on the Ten Gigabit Ethernet SPA installed in the top (0) subslot of the SIP that is located in slot 7 of the Cisco 7600 series router:

```
Router# show interfaces tengigabitethernet 7/0/0
TenGigabitEthernet7/0/0 is up, line protocol is up (connected)
  Hardware is TenGigEther SPA, address is 0000.0c00.0102 (bia 000f.342f.c340)
  Internet address is 15.1.1.2/24
  MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
```

Using the Cisco IOS Event Tracer to Troubleshoot Problems

Note This feature is intended for use as a software diagnostic tool and should be configured only under the direction of a Cisco Technical Assistance Center (TAC) representative.

The Event Tracer feature provides a binary trace facility for troubleshooting Cisco IOS software. This feature gives Cisco service representatives additional insight into the operation of the Cisco IOS software and can be useful in helping to diagnose problems in the unlikely event of an operating system malfunction or, in the case of redundant systems, Route Processor switchover.

Event tracing works by reading informational messages from specific Cisco IOS software subsystem components that have been preprogrammed to work with event tracing, and by logging messages from those components into system memory. Trace messages stored in memory can be displayed on the screen or saved to a file for later analysis.

The SPAs currently support the "spa" component to trace SPA OIR-related events.

For more information about using the Event Tracer feature, refer to the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide09186a0080087164.html

Preparing for Online Insertion and Removal of a SPA

The Cisco 7600 series router supports online insertion and removal (OIR) of the SIP, in addition to each of the SPAs. Therefore, you can remove a SIP with its SPAs still intact, or you can remove a SPA independently from the SIP, leaving the SIP installed in the router.

This means that a SIP can remain installed in the router with one SPA remaining active, while you remove another SPA from one of the SIP subslots. If you are not planning to immediately replace a SPA into the SIP, then be sure to install a blank filler plate in the subslot. The SIP should always be fully installed with either functional SPAs or blank filler plates.

For more information about activating and deactivating SPAs in preparation for OIR, see the "Preparing for Online Insertion and Removal of SIPs and SPAs" topic in the "Troubleshooting a SIP" chapter in this guide.

PART 6

Packet over SONET Shared Port Adapters

CHAPTER 14

Overview of the POS SPAs

This chapter provides an overview of the release history, and feature and Management Information Base (MIB) support for the Packet over SONET (POS) SPAs on the Cisco 7600 series router.

This chapter includes the following sections:

Release History

Release	Modification
Cisco IOS Release 12.2(33)SRA	Support for the following hardware was introduced on the Cisco 7600 series router: • The 2-Port and 4-Port OC-48c/STM-16 POS SPA was introduced on the Cisco 7600 SIP-600. • The 1-Port OC-48c/STM-16 POS SPA was introduced on the Cisco 7600 SIP-400.
Cisco IOS Release 12.2(18)SXF2	Support for the 1-Port OC-192c/STM-64 POS/RPR VSR Optics SPA was introduced on the Cisco 7600 SIP-600 on the Cisco 7600 series router and Catalyst 6500 series switch.

Cisco IOS Release 12.2(18)SXF	Support for the following hardware was introduced on the Cisco 7600 series router and Catalyst 6500 series switch: • 1-Port OC-192c/STM-64 POS/RPR SPA • 1-Port OC-192c/STM-64 POS/RPR XFP SPA
Cisco IOS Release 12.2(18)SXE	Support for the following hardware was introduced on the Cisco 7600 series router and Catalyst 6500 series switch: • 2-Port OC-3c/STM-1 POS SPA • 4-Port OC-3c/STM-1 POS SPA • 1-Port OC-12c/STM-4 POS SPA

POS Technology Overview

Packet-over-SONET is a high-speed method of transporting IP traffic between two points. This technology combines the Point-to-Point Protocol (PPP) with Synchronous Optical Network (SONET) and Synchronous Digital Hierarchy (SDH) interfaces.

SONET is an octet-synchronous multiplex scheme defined by the American National Standards Institute (ANSI) standard (T1.1051988) for optical digital transmission at hierarchical rates from 51.840 Mbps to 2.5 Gbps (Synchronous Transport Signal, STS-1 to STS-48) and greater. SDH is an equivalent international standard for optical digital transmission at hierarchical rates from 155.520 Mbps (Synchronous Transfer Mode-1 [STM-1]) to 2.5 Gbps (STM-16) and greater.

SONET specifications have been defined for single-mode fiber and multimode fiber. The POS SPAs on the Cisco 7600 series router allow transmission over both single-mode and multimode fiber at various optical carrier rates.

SONET/SDH transmission rates are integral multiples of 51.840 Mbps. The following transmission multiples are currently specified and used on the POS SPAs on the Cisco 7600 series router:

- OC-3c/STM-1—155.520 Mbps
- OC-12c/STM-4—622.080 Mbps
- OC-48—2.488 Gbps
- OC-192c/STM-64—9.953 Gbps

Supported Features

This section provides a list of some of the primary features supported by the POS SPA hardware and software:

- Jumbo frames (up to 9216 bytes)
- Online insertion and removal (OIR) from the SIP, or OIR of the SIP with the SPA inserted.
- Small form-factor pluggable (SFP) optics module OIR
- Field-programmable gate array (FPGA) upgrade support

The POS SPAs also support the following groups of features:

SONET/SDH Compliance Features

This section lists the SONET/SDH compliance features supported by the POS SPAs on the Cisco 7600 series router:

- 1+1 SONET Automatic Protection Switching (APS) as per G.783 Annex A
- 1+1 SDH Multiplex Section Protection (MSP) as per G.783 Annex A
- American National Standards Institute (ANSI) T1.105
- ITU-T G.707, G.783, G.957, G.958
- Telcordia GR-253-CORE: SONET Transport Systems: Common Generic Criteria
- Telcordia GR-1244: Clocks for the Synchronized Network: Common Generic Criteria

SONET/SDH Error, Alarm, and Performance Monitoring Features

This section lists the SONET/SDH error, alarm, and performance monitoring features supported by the POS SPAs on the Cisco 7600 series router:

- Signal failure bit error rate (SF-BER)
- Signal degrade bit error rate (SD-BER)
- Signal label payload construction (C2)
- Path trace byte (J1)
- Section:
  - Loss of signal (LOS)
  - Loss of frame (LOF)
  - Error counts for B1
  - Threshold crossing alarms (TCA) for B1
- Line:
  - Line alarm indication signal (LAIS)
  - Line remote defect indication (LRDI)
  - Line remote error indication (LREI)
  - Error counts for B2
  - Threshold crossing alarms (TCA) for B2
- Path:
  - Path alarm indication signal (PAIS)
  - Path remote defect indication (PRDI)

- Path remote error indication (PREI)
- Error counts for B3
- Threshold crossing alarms (TCA) for B3
- Loss of pointer (LOP)
- New pointer events (NEWPTR)
- Positive stuffing event (PSE)
- Negative stuffing event (NSE)

SONET/SDH Synchronization Features

This section lists the SONET/SDH synchronization features supported by the POS SPAs on the Cisco 7600 series router:

- Local (internal) timing (for inter-router connections over dark fiber or Wavelength Division Multiplex [WDM] equipment)
- Loop (line) timing (for connecting to SONET/SDH equipment)
- +/– 20 ppm clock accuracy over full operating temperature

WAN Protocol Features

This section lists the WAN protocols supported by the POS SPAs on the Cisco 7600 series router:

- RFC 1661, *The Point-to-Point Protocol (PPP)*
- RFC 1662, *PPP in HDLC framing*
- RFC 2615, *PPP over SONET/SDH* (with 1+x43 self-synchronous payload scrambling)
- RFC 3518, *Point-to-Point Protocol (PPP) Bridging Control Protocol (BCP)*—See Table 14-1 for BCP feature restrictions on the Cisco 7600 series router
- Cisco Protect Group Protocol over UDP/IP (Port 1972) for APS and MSP
- Multiprotocol Label Switching (MPLS)

Network Management Features

This section lists the network management features supported by the POS SPAs on the Cisco 7600 series router:

- Simple Network Management Protocol (SNMP) Management Information Base (MIB) counters
- Local (diagnostic) loopback
- Network loopback
- NetFlow Data Export
- IP over the Section Data Communications Channel (SDCC)—See Table 14-1 for SDCC feature restrictions on the Cisco 7600 series router
- RFC 3592 performance statistics for timed intervals (current, 15-minute, multiple 15-minute, and 1-day intervals):

- Regenerator section
- Multiplex section
- Path errored seconds
- Severely errored seconds
- Severely errored framed seconds

Restrictions

Note For other SIP-specific features and restrictions see also Chapter 3, "Overview of the SIPs and SSC."

Table 14-1 provides information about POS feature compatibility and restrictions by SIP and SPA combination.

Table 14-1 POS Feature Compatibility and Restrictions by SIP and SPA Combination

Feature	Cisco 7600 SIP-200	Cisco 7600 SIP-400	Cisco 7600 SIP-600
Bridge Control Protocol (BCP)	2-Port and 4-Port OC-3c/STM-1 POS SPA—Supported.	• 1-Port OC-12c/STM-4 POS SPA—Supported. • 2-Port and 4-Port OC-3c/STM-1 POS SPA—Supported. • 1-Port OC-48c/STM-16 POS SPA—Supported.	Not supported on any POS SPAs.
Dynamic Packet Transport (DPT), which includes RPR/SRP	Not supported on any POS SPAs.	Not supported on any POS SPAs.	Not supported on any POS SPAs.
Frame Relay	Supported on all POS SPAs.	Supported on all POS SPAs.	Not supported on any POS SPAs.
Section Data Communications Channel (SDCC)	• 2-Port OC-3c/STM-1 POS SPA—Supported. • 4-Port OC-3c/STM-1 POS SPA—SDCC is supported on up to two ports.	• 2-Port OC-3c/STM-1 POS SPA—Supported. • 4-Port OC-3c/STM-1 POS SPA—SDCC is supported on up to two ports. • 1-Port OC-12c/STM-4 POS SPA—Supported. • 1-Port OC-48c/STM-16 POS SPA—Not supported.	Not supported on any POS SPAs.

Supported MIBs

The following MIBs are supported in Cisco IOS Release 12.2(18)SXF2 for the 2-Port and 4-Port OC-3c/STM-1 POS SPA, 1-Port OC-12c/STM-4 POS SPA, 1-Port OC-192c/STM-64 POS/RPR SPA, 1-Port OC-192c/STM-64 POS/RPR XFP SPA, and 1-Port OC-192c/STM-64 POS/RPR VSR Optics SPA on the Cisco 7600 series router:

- CISCO-APS-MIB
- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- CISCO-ENVMON-MIB (For NPEs, NSEs, line cards, and MSCs only)
- CISCO-EXTENDED-ENTITY-MIB
- CISCO-OPTICAL-MIB
- ENTITY-MIB
- OLD-CISCO-CHASSIS-MIB
- IF-MIB
- SONET-MIB (RFC 2558, *Definitions of Managed Objects for SONET/SDH Interface Type*)

For more information about MIB support on Cisco xxxx series routers, refer to the *Cisco 7600 Series Internet Router MIB Specifications Guide*, at the following URL:

http://www.cisco.com/en/US/products/hw/routers/ps368/prod_technical_reference_list.html

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

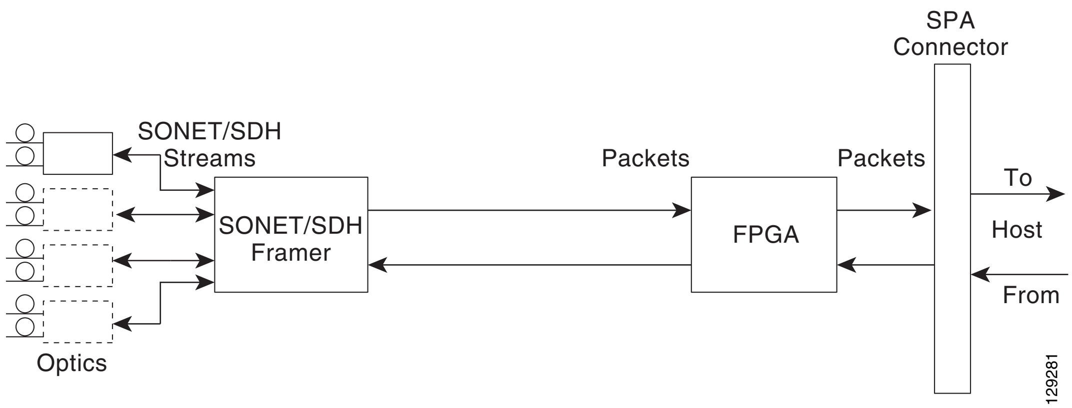
SPA Architecture

This section provides an overview of the architecture of the POS SPAs and describes the path of a packet in the ingress and egress directions. Some of these areas of the architecture are referenced in the SPA software and can be helpful to understand when troubleshooting or interpreting some of the SPA CLI and **show** command output.

4-Port OC-3c/STM-1 POS SPA Architecture

Figure 14-1 identifies some of the hardware devices that are part of the POS SPA architecture. The figure shows the four ports that are supported by the 4-Port OC-3c/STM-1 POS SPA only.

Figure 14-1 4-Port OC-3c/STM-1 POS SPA Architecture



Every incoming and outgoing packet on the 4-Port OC-3c/STM-1 POS SPA goes through the SONET/SDH framer and field-programmable gate array (FPGA) devices.

Path of a Packet in the Ingress Direction

The following steps describe the path of an ingress packet through the 4-Port OC-3c/STM-1 POS SPA:

1. The framer receives SONET/SDH streams from the SFP optics, extracts clocking and data, and processes the section, line, and path overhead.
2. The framer extracts the POS frame payload and verifies the frame size and frame check sequence (FCS).
3. The framer passes valid frames to the field-programmable gate array (FPGA) on the SPA.
4. The FPGA on the SPA transfers frames to the host through the SPI4.2 bus for further processing and switching.

Path of a Packet in the Egress Direction

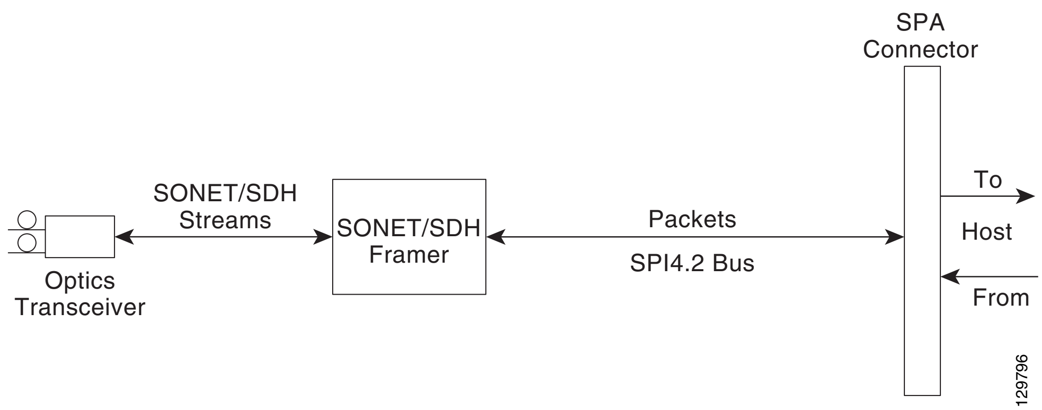
The following steps describe the path of an egress packet through the 4-Port OC-3c/STM-1 POS SPA:

1. The host sends packets to the FPGA on the SPA using the SPI4.2 bus.
2. The FPGA on the SPA stores the data in the appropriate channel's first-in first-out (FIFO) queue.
3. The FPGA on the SPA passes the packet to the framer.
4. The framer accepts the data and stores it in the appropriate channel queue.
5. The framer adds the FCS and SONET/SDH overhead.
6. The framer sends the data to the SFP optics for transmission onto the network.

1-Port OC-192c/STM-64 POS/RPR XFP SPA Architecture

Figure 14-2 identifies the primary hardware devices that are part of the POS SPA architecture. The figure shows a single optics transceiver supported by both of the POS SPAs. However, the 1-Port OC-192c/STM-64 POS/RPR SPA and 1-Port OC-192c/STM-64 POS/RPR VSR Optics SPA support fixed optics, while the 1-Port OC-192c/STM-64 POS/RPR XFP SPA supports XFP optics. The path of a packet remains the same except for where the optic transceiver support resides.

Figure 14-2 1-Port OC-192c/STM-64 POS/RPR XFP SPA Architecture



In POS mode, every incoming and outgoing packet on the OC-192 POS SPAs goes through the SONET/SDH framer and SPI4.2 interface.

Path of a Packet in the Ingress Direction

The following steps describe the path of an ingress packet through the 1-Port OC-192c/STM-64 POS/RPR XFP SPA:

1. The framer receives SONET/SDH streams from the XFP optics, extracts clocking and data, and processes the section, line, and path overhead.
2. The framer extracts the POS frame payload and verifies the frame size and frame check sequence (FCS).
3. The framer passes valid frames to the System Packet Level Interface 4.2 (SPI4.2) interface on the SPA.
4. The SPI4.2 interface transfers frames to the host through the SPI4.2 bus for further processing and switching.

Path of a Packet in the Egress Direction

The following steps describe the path of an egress packet through the 1-Port OC-192c/STM-64 POS/RPR XFP SPA:

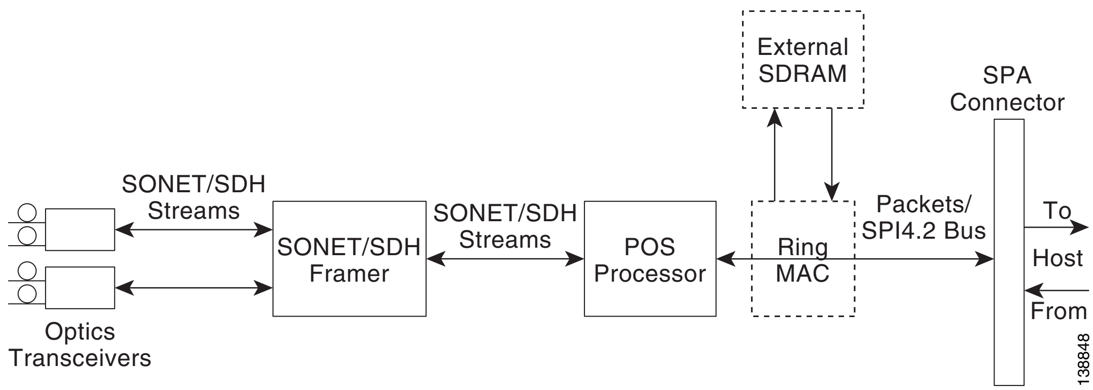
1. The host sends packets to the SPA using the SPI4.2 bus.
2. The SPA stores the data in the appropriate channel's first-in first-out (FIFO) queue.
3. The SPA passes the packet to the framer.
4. The framer accepts the data and stores it in the appropriate channel queue.

5. The framer adds the FCS and SONET/SDH overhead.
6. The framer sends the data to the XFP optics for transmission onto the network.

2-Port OC-48c/STM-16 POS SPA Architecture

Figure 14-3 identifies the primary hardware devices that are part of the 2-Port OC-48c/STM-16 POS SPA architecture.

Figure 14-3 2-Port OC-48c/STM-16 POS SPA Architecture



Path of a Packet in the Ingress Direction

The following steps describe the path of an ingress packet through the 2-Port OC-48c/STM-16 POS SPA:

1. The framer receives SONET/SDH streams from the SFP optics, extracts clocking and data, and processes the section, line, and path overhead.
2. The framer detects Loss of Signal (LOS), Loss of Frame (LOF), Severely Errored Frame (SEF), Line Alarm Indication Signal (AIS-L), Loss of Pointer (LOP), Line Remote Defect Indication Signal (Enhanced RDI-L), Path Alarm Indication Signal (AIS-P), Standard and Enhanced Path Remote Defect Indication Signal (RDI-P), Path Remote Error Indication (Enhanced REI-P). The framer extracts or inserts DCC bytes.
3. The framer processes the S1 synchronization status byte, the pointer action bytes (per Telcordia GR-253-CORE), and extracts or inserts DCC bytes.
4. The POS processor extracts the POS frame payload and verifies the frame size and frame check sequence (FCS).
5. The POS processor supports PPP, Frame Relay, or HDLC modes and optionally performs payload scrambling.
6. The POS processor passes valid frames to the System Packet Level Interface 4.2 (SPI4.2) interface on the SPA.
7. The SPI4.2 interface transfers frames to the host through the SPI4.2 bus for further processing and switching.

Path of a Packet in the Egress Direction

The following steps describe the path of an egress packet through the 2-Port OC-48c/STM-16 POS SPA:

1. The host sends packets to the SPA using the SPI4.2 bus.
2. The SPA stores the data in the appropriate SPI4 channel's first-in first-out (FIFO) queue.
3. The SPA passes the packet from the SPI4 interface to the POS processor where it is encapsulated in a POS frame and FCS is added.
4. The POS frame is sent to the SONET/SDH framer where it is placed into the SONET payload.
5. The framer adds the FCS and SONET/SDH overhead.
6. The framer sends the data to the SFP optics for transmission onto the network.

Displaying the SPA Hardware Type

To verify the SPA hardware type that is installed in your Cisco 7600 series router, you can use the **show idprom** command. For other hardware information, you can also use the **show interfaces** or **show controllers** commands. There are several other commands on the Cisco 7600 series router that also provide SPA hardware information. For more information about these commands, see the "Command Summary for POS SPAs" and the "SIP and SPA Commands" chapters in this guide.

Table 14-2 shows the hardware description that appears in the **show** command output for each type of SPA that is supported on the Cisco 7600 series router.

Table 14-2 SPA Hardware Descriptions in show Commands

SPA	Description in show interfaces Command	Description in show idprom Command
2-Port OC-3c/STM-1 POS SPA	Hardware is Packet over Sonet	2-port OC3/STM1 POS Shared Port Adapter / SPA-2XOC3-POS
4-Port OC-3c/STM-1 POS SPA	Hardware is Packet over Sonet	4-port OC3/STM1 POS Shared Port Adapter / SPA-4XOC3-POS
1-Port OC-12c/STM-4 POS SPA	Hardware is Packet over Sonet	1-port OC12/STM4 POS Shared Port Adapter / SPA-1XOC12-POS
1-Port OC-48c/STM-16 POS SPA	Hardware is Packet over Sonet	1-port OC48/STM16 POS/RPR Shared Port Adapter / SPA-1XOC48POS/RPR
2-Port OC-48c/STM-16 POS SPA	Hardware is Packet over Sonet	2-port OC48/STM16 POS/RPR Shared Port Adapter / SPA-2XOC48POS/RPR
4-Port OC-48c/STM-16 POS SPA	Hardware is Packet over Sonet	4-port OC48/STM16 POS/RPR Shared Port Adapter / SPA-4XOC48POS/RPR

Table 14-2 SPA Hardware Descriptions in show Commands (continued)

SPA	Description in show interfaces Command	Description in show idprom Command
1-Port OC-192c/STM-64 POS/RPR SPA	Hardware is Packet over Sonet	1-port OC192/STM64 POS/RPR Shared Port Adapter / SPA-OC192POS-VSR / SPA-OC192POS-LR
1-Port OC-192c/STM-64 POS/RPR XFP SPA	Hardware is Packet over Sonet	1-port OC192/STM64 POS/RPR XFP Optics Shared Port Adapter / SPA-OC192POS-XFP

Example of the show idprom Command

The following example shows sample output for the **show idprom module detail** command for a 4-Port OC-3c/STM-1 POS SPA installed in subslot 3 of the SIP installed in slot 2 of the router:

```
Router# show idprom module 2/3 detail
IDPROM for SPA module #2/3
        (FRU is '4-port OC3/STM1 POS Shared Port Adapter')
        EEPROM version           : 4
        Compatible Type          : 0xFF
        Controller Type          : 1088
        Hardware Revision        : 0.230
        Boot Timeout             : 0 msecs
        PCB Serial Number        : PRTA0304155
        Part Number              : 73-9313-02
        73/68 Board Revision     : 04
        Fab Version              : 02
        RMA Test History         : 00
        RMA Number               : 0-0-0-0
        RMA History              : 00
        Deviation Number         : 0
        Product Identifier (PID) : SPA-4XOC3-POS
        Version Identifier (VID) : V01
.
.
.
```

Example of the show interfaces Command

The following example shows output from the **show interfaces pos** command on a Cisco 7600 series router with a 4-Port OC-3c/STM-1 POS SPA installed in slot 5:

```
Router# show interfaces pos 5/0/1
POS5/0/1 is up, line protocol is up
  Hardware is Packet over Sonet
  Internet address is 10.5.5.5/8
  MTU 4470 bytes, BW 155000 Kbit, DLY 100 usec,
     reliability 96/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive not set
  Scramble disabled
  Last input 00:00:11, output 00:00:11, output hang never
  Last clearing of ''show interface'' counters 00:00:23
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
```

```
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  5 packets input, 520 bytes
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  5 packets output, 520 bytes, 0 underruns
     0 output errors, 0 applique, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

Example of the show controllers Command

The following example shows output from the **show controllers pos** command on a Cisco 7600 series router for the first interface (0) of a POS SPA installed in subslot 2 of a SIP installed in chassis slot 3:

```
Router# show controllers pos 3/2/0
POS3/2/0
SECTION
LOF = 0 LOS = 0 BIP(B1) = 0
LINE
AIS = 0 RDI = 0 FEBE = 0 BIP(B2) = 0
PATH
AIS = 0 RDI = 0 FEBE = 0 BIP(B3) = 0
PLM = 0 UNEQ = 0 TIM = 0 TIU = 0
LOP = 0 NEWPTR = 0 PSE = 0 NSE = 0

Active Defects: None
Active Alarms: None
Alarm reporting enabled for: SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA

Framing: SONET
APS

COAPS = 0 PSBF = 0
State: PSBF_state = False
Rx(K1/K2): 00/00 Tx(K1/K2): 00/00
Rx Synchronization Status S1 = 00
S1S0 = 00, C2 = CF
Remote aps status (none); Reflected local aps status (none)
CLOCK RECOVERY
RDOOL = 0
State: RDOOL_state = False
PATH TRACE BUFFER: STABLE
Remote hostname : sip-sw-7600-2
Remote interface: POS3/2/1
Remote IP addr : 0.0.0.0
Remote Rx(K1/K2): 00/00 Tx(K1/K2): 00/00

BER thresholds: SF = 10e-3 SD = 10e-6
TCA thresholds: B1 = 10e-6 B2 = 10e-6 B3 = 10e-6

Clock source: internal
```

CHAPTER 15

Configuring the POS SPAs

This chapter provides information about configuring the Packet over SONET (POS) shared port adapters (SPAs) on the Cisco 7600 series router. This chapter includes the following sections:

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications that correspond to your Cisco IOS software release.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Configuration Tasks

This section describes how to configure POS SPAs and includes information about verifying the configuration.

It includes the following topics:

- Shutting Down and Restarting an Interface on a SPA, page 15-14

Specifying the Interface Address on a SPA

SPA interface ports begin numbering with "0" from left to right. Single-port SPAs use only the port number 0. To configure or monitor SPA interfaces, you need to specify the physical location of the SIP, SPA, and interface in the CLI. The interface address format is *slot*/*subslot*/*port*, where:

- *slot*—Specifies the chassis slot number in the Cisco 7600 series router where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA.

The following example shows how to specify the first interface (0) on a SPA installed in the first subslot of a SIP (0) installed in chassis slot 3:

```
Router(config)# interface serial 3/0/0
```

This command shows a serial SPA as a representative example, however the same *slot*/*subslot*/*port* format is similarly used for other SPAs (such as ATM and POS) and other non-channelized SPAs.

Modifying the Interface MTU Size

The Cisco IOS software supports three different types of configurable maximum transmission unit (MTU) options at different levels of the protocol stack:

- Interface MTU—Checked by the SPA on traffic coming in from the network. Different interface types support different interface MTU sizes and defaults. The interface MTU defines the maximum packet size allowable (in bytes) for an interface before drops occur. If the frame is smaller than the interface MTU size, but is not smaller than three bytes of payload size, then the frame continues to process.
- IP MTU—Can be configured on a subinterface and is used by the Cisco IOS software to determine whether fragmentation of a packet takes place. If an IP packet exceeds the IP MTU size, then the packet is fragmented.
- Tag or Multiprotocol Label Switching (MPLS) MTU—Can be configured on a subinterface and allows up to six different labels, or tag headers, to be attached to a packet. The maximum number of labels is dependent on your Cisco IOS software release.

Different encapsulation methods and the number of MPLS MTU labels add additional overhead to a packet. For example, for an Ethernet packet, SNAP encapsulation adds an 8-byte header, dot1q encapsulation adds a 2-byte header, and each MPLS label adds a 4-byte header (*n* labels x 4 bytes).

Interface MTU Configuration Guidelines

When configuring the interface MTU size on the POS SPAs, consider the following guidelines:

- If you are also using MPLS, be sure that the **mpls mtu** command is configured for a value less than or equal to the interface MTU.
- If you change the interface MTU size, the giant counter increments when the interface receives a packet that exceeds the MTU size that you configured, plus an additional 88 bytes for overhead, and an additional 2 or 4 bytes for the configured cyclic redundancy check (CRC).

  For example, with a maximum MTU size of 9216 bytes, the giant counter increments:

- For a 16-bit CRC (or FCS), when receiving packets larger than 9306 bytes (9216 + 88 + 2).
- For a 32-bit CRC, when receiving packets larger than 9308 bytes (9216 + 88 + 4).

- The Frame Relay Local Management Interface (LMI) protocol requires that all permanent virtual circuit (PVC) status reports fit into a single packet. Using the default MTU of 4470 bytes, this limits the number of data-link connection identifiers (DLCIs) to 890. The following formula demonstrates how to determine the maximum DLCIs for a configured interface MTU:
  - Maximum DLCIs = (MTU bytes – 20)/(5 bytes per DLCI)
  - Maximum DLCIs for the default MTU = (4470 – 20)/5 = 890 DLCIs per interface

Interface MTU Configuration Task

To modify the MTU size on an interface, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **mtu** *bytes*	Configures the maximum packet size for an interface, where: • *bytes*—Specifies the maximum number of bytes for a packet. The default is 4470 bytes.

To return to the default MTU size, use the **no** form of the command.

Verifying the MTU Size

To verify the MTU size for an interface, use the **show interfaces pos** privileged EXEC command and observe the value shown in the "MTU" field.

The following example shows an MTU size of 4470 bytes for interface port 0 (the first port) on the SPA installed in subslot 1 of the SIP that is located in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces pos 2/1/0
POS2/1/0 is up, line protocol is up (APS working - active)
  Hardware is Packet over Sonet
  Internet address is 10.1.1.1/24
  MTU 4470 bytes, BW 155000 Kbit, DLY 100 usec,
    reliability 255/255, txload 1/255, rxload 1/255.
.
```

Modifying the POS Framing

POS framing can be specified as SONET (Synchronous Optical Network) or SDH (Synchronous Digital Hierarchy). SONET and SDH are a set of related standards for synchronous data transmission over fiber-optic networks. SONET is the United States version of the standard published by the American National Standards Institute (ANSI). SDH is the international version of the standard published by the International Telecommunications Union (ITU).

To modify the POS framing, use the following command in interface configuration mode:

To return to the default, use the **no** form of the command.

Verifying the POS Framing

To verify the POS framing, use the **show controllers pos** privileged EXEC command and observe the value shown in the "Framing" field. The following example shows that POS framing mode is set to SONET for the first interface (0) on the POS SPA installed in subslot 2 of a SIP installed in chassis slot 3:

```
Router# show controllers pos 3/2/0
POS3/2/0
SECTION
LOF = 0 LOS = 0 BIP(B1) = 0
LINE
AIS = 0 RDI = 0 FEBE = 0 BIP(B2) = 0
PATH
AIS = 0 RDI = 0 FEBE = 0 BIP(B3) = 0
PLM = 0 UNEQ = 0 TIM = 0 TIU = 0
LOP = 0 NEWPTR = 0 PSE = 0 NSE = 0

Active Defects: None
Active Alarms: None
Alarm reporting enabled for: SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA

Framing: SONET
APS

COAPS = 0 PSBF = 0
State: PSBF_state = False
Rx(K1/K2): 00/00 Tx(K1/K2): 00/00
Rx Synchronization Status S1 = 00
S1S0 = 00, C2 = CF
Remote aps status (none); Reflected local aps status (none)
CLOCK RECOVERY
RDOOL = 0
State: RDOOL_state = False
PATH TRACE BUFFER: STABLE
Remote hostname : sip-sw-7600-2
Remote interface: POS3/2/1
Remote IP addr : 0.0.0.0
Remote Rx(K1/K2): 00/00 Tx(K1/K2): 00/00

BER thresholds: SF = 10e-3 SD = 10e-6
TCA thresholds: B1 = 10e-6 B2 = 10e-6 B3 = 10e-6

Clock source: internal
```

Modifying the Keepalive Interval

When the keepalive feature is enabled, a keepalive packet is sent at the specified time interval to keep the interface active. The keepalive interval must be configured to be the same on both ends of the POS link.

To modify the keepalive interval, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **keepalive** [*period* [*retries*]]	Specifies the frequency at which the Cisco IOS software sends messages to the other end of the link, to ensure that a network interface is alive, where: • *period*—Specifies the time interval in seconds for sending keepalive packets. The default is 10 seconds. • *retries*—Specifies the number of times that the device will continue to send keepalive packets without response before bringing the interface down. The default is 5 retries.

To disable keepalive packets, use the **no** form of this command.

Note If keepalives are enabled and you are trying to configure line loopback on a POS interface, the keepalive protocol will fail and periodically reset the interface based on the keepalive timeout and cause Layer 1 errors on the other end of the link that is trying to do the loopbacks.

You can avoid this by using the **no keepalive** command on the POS interface that is configured for line loopback. The side that is not in line loopback detects that its keepalive is being looped back and functions properly. An interface configured for internal loopback also functions properly with keepalives enabled.

Verifying the Keepalive Interval

To verify the keepalive interval, use the **show interfaces pos** privileged EXEC command and observe the value shown in the "Keepalive" field.

The following example shows that keepalive is enabled for interface port 0 on the POS SPA installed in the SIP that is located in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces pos 2/0/0
    Hardware is Packet over Sonet
    Internet address is 10.1.1.1.2
    MTU 9216 bytes, BW 622000 Kbit, DLY 100 usec, reliability 255/255, txload 1/255,
    rxload 1/255
      Keepalive set (10 sec)
.
.
.
```

Modifying the CRC Size

CRC is an error-checking technique that uses a calculated numeric value to detect errors in transmitted data. The CRC size indicates the length in bits of the FCS.

The CRC size must be configured to be the same on both ends of the POS link.

To modify the CRC size, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **crc** [**16** | **32**]	(As Required) Specifies the length of the cyclic redundancy check (CRC), where: • **16**—Specifies a 16-bit length CRC. This is the default. • **32**—Specifies a 32-bit length CRC. The CRC size must be configured to be the same on both ends of the POS link.

To return to the default CRC size, use the **no** form of the command.

Verifying the CRC Size

To verify the CRC size, use the **show interfaces pos** privileged EXEC command and observe the value shown in the "CRC" field.

The following example shows that the CRC size is 16 for interface port 0 on the POS SPA installed in the SIP that is located in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces pos 2/0/0
    Hardware is Packet over Sonet
    Internet address is 10.1.1.2.1
    MTU 9216 bytes, BW 622000 Kbit, DLY 100 usec reliability 255/255, txload 1/255, rxload
    1/255
        Encapsulation HDLC, crc 16, loopback not set
.
.
.
```

Modifying the Clock Source

A clock source of internal specifies that the interface clocks its transmitted data from its internal clock. A clock source of line specifies that the interface clocks its transmitted data from a clock recovered from the line's receive data stream.

For information about the recommended clock source settings for POS router interfaces, refer to *Configuring Clock Settings on POS Router Interfaces* at the following URL:

http://www.cisco.com/en/US/tech/tk482/tk607/technologies_tech_note09186a0080094bb9.shtml

To modify the clock source, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **clock source** {**line** | **internal**}	Specifies the clock source for the POS link, where: • **line**—The link uses the recovered clock from the line. This is the default. • **internal**—The link uses the internal clock source.

To return to the default clock source, use the **no** form of this command.

Verifying the Clock Source

To verify the clock source, use the **show controllers pos** privileged EXEC command and observe the value shown in the "Clock source" field.

The following example shows that the clock source is internal for interface port 0 on the POS SPA installed in subslot 0 of the SIP that is located in slot 2 of the Cisco 7600 series router:

```
Router# show controllers pos 2/0/0
POS2/0/0
SECTION
LOF = 0 LOS = 1 BIP(B1) = 7
LINE
AIS = 0 RDI = 1 FEBE = 20 BIP(B2) = 9
PATH
AIS = 0 RDI = 0 FEBE = 0 BIP(B3) = 5
PLM = 0 UNEQ = 0 TIM = 0 TIU = 0
LOP = 0 NEWPTR = 0 PSE = 0 NSE = 0


Active Defects: None
Active Alarms: None

Alarm reporting enabled for: SF SLOS SLOF B1-TCA LAIS LRDI B2-TCA PAIS PLOP PRDI PUNEQ
B3-TCA RDOOL

APS

COAPS = 2 PSBF = 0
State: PSBF_state = False
Rx(K1/K2): 00/00 Tx(K1/K2): 00/00
Rx Synchronization Status S1 = 00
S1S0 = 02, C2 = CF
CLOCK RECOVERY
RDOOL = 0
State: RDOOL_state = False
PATH TRACE BUFFER: STABLE
Remote hostname : RouterTester. Port 102/1
Remote interface:
Remote IP addr :
Remote Rx(K1/K2): / Tx(K1/K2): /

BER thresholds: SF = 10e-5 SD = 10e-6
```

```
TCA thresholds: B1 = 10e-6 B2 = 10e-6 B3 = 10e-6

Clock source: internal
.
```

Modifying SONET Payload Scrambling

SONET payload scrambling applies a self-synchronous scrambler (x43+1) to the Synchronous Payload Envelope (SPE) of the interface to ensure sufficient bit transition density.

The default configuration is SONET payload scrambling disabled.

SONET payload scrambling must be configured to be the same on both ends of the POS link.

To modify SONET payload scrambling, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **pos scramble-atm**	Enables SONET payload scrambling.

To disable SONET payload scrambling, use the **no** form of this command.

Verifying SONET Payload Scrambling

To verify SONET payload scrambling, use the **show interfaces pos** privileged EXEC command and observe the value shown in the “Scramble” field.

The following example shows that SONET payload scrambling is disabled for interface port 0 on the POS SPA installed in subslot 0 of the SIP that is located in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces pos 2/0/0
    Hardware is Packet over Sonet
    Internet address is 10.0.0.1/24
    MTU 9216 bytes, BW 622000 Kbit, DLY 100 usec,
        reliability 255/255, txload 1/255, rxload 1/255
        Encapsulation HDLC, crc 16, loopback not set
    Keepalive not set
    Scramble disabled
.
```

Configuring the Encapsulation Type

By default, the POS interfaces support High-Level Data Link Control (HDLC) encapsulation. The encapsulation method can be specified as HDLC, Point-to-Point Protocol (PPP) or Frame Relay. The encapsulation type must be configured to be the same on both ends of the POS link.

To modify the encapsulation method, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **encapsulation** *encapsulation-type*	Specifies the encapsulation method used by the interface, where: • *encapsulation-type*—Can be HDLC, PPP, or Frame Relay. The default is HDLC.

Verifying the Encapsulation Type

To verify the encapsulation type, use the **show interfaces pos** privileged EXEC command and observe the value shown in the "Encapsulation" field.

The following example shows the encapsulation type is HDLC for port 0 on the POS SPA installed in subslot 0 of the SIP that is located in slot 2 of the Cisco 7600 series router:

```
Router# show interfaces pos 2/0/0
  Hardware is Packet over Sonet
  Internet address is 10.0.0.1/24
  MTU 9216 bytes, BW 622000 Kbit, DLY 100 usec,
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
    Keepalive not set
    Scramble disabled
.
.
.
```

Configuring APS

Automatic protection switching (APS) allows switchover of POS circuits in the event of circuit failure and is often required when connecting SONET equipment to telco equipment. APS refers to the mechanism of using a "protect" POS interface in the SONET network as the backup for a "working" POS interface. When the working interface fails, the protect interface quickly assumes its traffic load. Depending on the configuration, the two circuits may be terminated in the same router, or in different routers.

For more information about APS, refer to *A Brief Overview of Packet Over SONET APS* at the following URL:

http://www.cisco.com/en/US/tech/tk482/tk607/technologies_tech_note09186a0080093eb5.shtml

To configure the working POS interface, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **aps working** *circuit-number*	Configures a POS interface as a working APS interface, where: • *circuit-number*—Specifies the circuit number associated with this working interface.

To remove the POS interface as a working interface, use the **no** form of this command.

To configure the protect POS interface, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **aps protect** *circuit-number ip-address*	Configures a POS interface as a protect APS interface, where: • *circuit-number*—Specifies the number of the circuit to enable as a protect interface. • *ip-address*—Specifies the IP address of the router that has the working POS interface.

To remove the POS interface as a protect interface, use the **no** form of this command.

Verifying the APS Configuration

To verify the APS configuration or to determine if a switchover has occurred, use the **show aps** command.

The following is an example of a router configured with a working interface. In this example, POS interface 0/0/0 is configured as a working interface in group 1, and the interface is selected (that is, active).

```
Router# show aps
POS0/0/0 working group 1 channel 1 Enabled Selected
```

The following is an example of a router configured with a protect interface. In this example, POS interface 2/1/1 is configured as a protect interface in group 1. The output also shows that the working channel is located on the router with the IP address 10.0.0.1 and that the interface currently selected is enabled.

```
Router# show aps
POS2/1/1 APS Group 1: protect channel 0 (inactive)
Working channel 1 at 10.0.0.1 (Enabled)
  SONET framing; SONET APS signalling by default
  Remote APS configuration: (null)
.
```

Configuring POS Alarm Trigger Delays

A trigger is an alarm that, when activated, causes the line protocol to go down. The POS alarm trigger delay helps to ensure uptime of a POS interface by preventing intermittent problems from disabling the line protocol. The POS alarm trigger delay feature delays the setting of the line protocol to down when trigger alarms are received. If the trigger alarm was sent because of an intermittent problem, the POS alarm trigger delay can prevent the line protocol from going down when the line protocol is functional.

Line-Level and Section-Level Triggers

The **pos delay triggers line** command is used for POS router interfaces connected to internally-protected Dense Wavelength Division Multiplexing (DWDM) systems. This command is invalid for interfaces that are configured as working or protect APS. Normally a few microseconds of line- or section-level alarms brings down the link until the alarm has been clear for ten seconds. If you configure holdoff, the link-down trigger is delayed for 100 milliseconds. If the alarm stays up for more than 100 milliseconds, the link is brought down. If the alarm clears before 100 milliseconds, the link remains up.

The following line- and section-level alarms are triggers, by default, for the line protocol to go down:

- Line alarm indication signal (LAIS)
- Section loss of signal (SLOS)
- Section loss of frame (SLOF)

You can issue the **pos delay triggers line** command to delay a down trigger of the line protocol on the interface. You can set the delay from 50 to 10000 milliseconds. The default delay is 100 milliseconds.

To configure POS line- or section-level triggers, use the following commands beginning in interface configuration mode:

	Command	Purpose
Step 1	Router(config-if)# **pos delay triggers line** *ms*	Specifies a delay for setting the line protocol to down when a line-level trigger alarm is received, where: • *ms*—Specifies the delay in milliseconds. The default delay is 100 milliseconds.
Step 2	Router(config-if)# **pos threshold** {**b1-tca** | **b2-tca** | **b3-tca** | **sd-ber** | **sf-ber**} *rate*	Configures the POS bit error rate (BER) threshold values of the specified alarms, where: • **b1-tca** *rate*—Specifies the B1 BER threshold crossing alarm. The default is 6. • **b2-tca** *rate*—Specifies the B2 BER threshold crossing alarm. The default is 6. • **b3-tca** *rate*—Specifies the B3 BER threshold crossing alarm. The default is 6. • **sd-ber** *rate*—Specifies the signal degrade BER threshold. The default is 6. • **sf-ber** *rate*—Specifies the signal failure BER threshold. The default is 3. • *rate*—Specifies the bit error rate from 3 to 9 (10e-n). The default varies by the type of threshold that you configure.
Step 3	Router(config-if)# **pos ais-shut**	Sends a line alarm indication signal (AIS-L) to the other end of the link after a **shutdown** command has been issued to the specified POS interface. AIS-L is also known as LAIS when alarm-related output is generated using the **show controllers pos** command. By default, the AIS-L is not sent to the other end of the link. Stops transmitting the AIS-L by issuing either the **no shutdown** or the **no pos ais-shut** commands.

To disable alarm trigger delays, use the **no** form of the **pos delay triggers line** command.

To determine which alarms are reported on the POS interface, and to display the BER thresholds, use the **show controllers pos** command.

Path-Level Triggers

You can issue the **pos delay triggers path** command to configure various path alarms as triggers and to specify an activation delay between 50 and 10000 milliseconds. The default delay value is 100 milliseconds. The following path alarms are not triggers by default. You can configure these path alarms as triggers and also specify a delay:

- Path alarm indication signal (PAIS)
- Path remote defect indication (PRDI)
- Path loss of pointer (PLOP)
- sd-ber (signal degrade [SD] bit error rate [BER])
- sf-ber (signal failure [SF] BER)
- b1-tca (B1 BER threshold crossing alarm [TCA])
- b2-tca (B2 BER TCA)
- b3-tca (B3 BER TCA)

The **pos delay triggers path** command can also bring down the line protocol when the higher of the B2 and B3 error rates is compared with the signal failure (SF) threshold. If the SF threshold is crossed, the line protocol of the interface goes down.

To configure POS path-level triggers, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **pos delay triggers path** *ms*	Specifies that path-level alarms should act as triggers and specifies a delay for setting the line protocol to down when a path-level trigger alarm is received, where: • *ms*—Specifies the delay in milliseconds. The default delay is 100 milliseconds.

To disable path-level triggers, use the **no** form of this command.

Verifying POS Alarm Trigger Delays

To verify POS alarm trigger delays, use the **show controllers pos** privileged EXEC command and observe the values shown in the "Line alarm trigger delay" and "Path alarm trigger delay" fields.

The following example shows the POS alarm trigger delays for interface port 0 on the POS SPA installed in the SIP that is located in slot 2 of the Cisco 7600 series router:

```
Router# show controllers pos 2/0/0 details
POS2/0/0
SECTION
LOF = 0 LOS = 1 BIP(B1) = 5
LINE
AIS = 0 RDI = 1 FEBE = 5790 BIP(B2) = 945
PATH
AIS = 0 RDI = 0 FEBE = 0 BIP(B3) = 5
PLM = 0 UNEQ = 0 TIM = 0 TIU = 0
LOP = 1 NEWPTR = 0 PSE = 0 NSE = 0

Active Defects: None
```

```
Active Alarms: None
Alarm reporting enabled for: SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA

Line alarm trigger delay = 100 ms
Path alarm trigger delay = 100 ms
.
.
.
```

Configuring SDCC

Before any management traffic can traverse the section data communication channel (SDCC) links embedded in the POS SPA overhead, the SDCC interfaces must be configured and activated.

SDCC Configuration Guidelines

When configuring SDCC on a POS SPA, consider the following guidelines:

- SDCC must be enabled on the main POS interfaces.
- SDCC supports only HDLC and PPP encapsulation, not Frame Relay.

SDCC Configuration Task

To configure the POS SPAs for SDCC, complete the following steps:

Verifying the SDCC Interface Configuration

To verify the SDCC interface, use the **show interfaces sdcc** privileged EXEC command and observe the value shown in the “Hardware is” field.

The following example shows the SDCC interface port 1 on the POS SPA installed in subslot 0 of the SIP that is located in slot 5 of the Cisco 7600 series router:

```
Router# show interfaces sdcc 5/0/1
SDCC5/0/1 is up, line protocol is up
  Hardware is SDCC
  Internet address is 10.14.14.14/8
  MTU 1500 bytes, BW 155000 Kbit, DLY 20000 usec,
     reliability 5/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive not set
  Last input 00:01:24, output never, output hang never
  Last clearing of ''show interface'' counters 00:01:30
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  5 packets input, 520 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  5 packets output, 520 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

- The default mode for all SPA interfaces is POS. To change between POS and SRP modes, you must shut down the SPA interface.
- Whenever you change modes on a POS SPA, the SPA automatically reloads.
- To change the SRP mate configuration, you must shut down the SPA interfaces.
- You cannot configure subinterfaces on an SRP interface.
- To distinguish between the two rings, one is referred to as the "inner" ring and the other as the "outer" ring. SRP operates by sending data packets in one direction (downstream) and sending the corresponding control packets in the opposite direction (upstream) on the other fiber. An SRP node uses SRP side A to receive (RX) outer ring data and transmit (TX) inner ring data. The node uses SRP side B to receive (RX) inner ring data and transmit (TX) outer ring data. Side A on one node connects to Side B on an adjacent SRP node.

  For configuration of SRP on POS SPAs in multiple slots on the same SIP, the lower-numbered slot and subslot combination hosts the SRP interface and becomes "Side A" of the SRP interface. The slot number of the side-A interface must be lower than the slot location of the SRP mate (side B) interface.
- To configure SRP options, you must specify the slot and subslot location of the side-A interface, in addition to a port number.

SRP Mode Configuration Guidelines

When enabling SRP mode, consider the following guidelines:

- **hw-module subslot srp** command You only need to configure the **hw-module subslot srp** command on the host SRP interface—not on the mate SRP interface.
- The host SRP interface becomes "Side A" of the SRP interface. When configuring SPAs that are installed in different slots on the same SIP for SRP, the slot number of the side-A interface must be lower than the slot location of the SRP mate (side B) interface. Also, you must specify the side-A interface location for configuration of any SRP options.
- The SIP reads the information it receives from the hardware cable mating to validate the mate cable connectivity with your software configuration.
- When you change the SPA mode, the SPA automatically reloads.

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), use the following command in privileged EXEC configuration mode:

Command	Purpose
Router# **copy running-config startup-config**	Writes the new configuration to NVRAM.

For more information about managing configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

Shutting Down and Restarting an Interface on a SPA

You can shut down and restart any of the interface ports on a SPA independently of each other. Shutting down an interface stops traffic and then enters the interface into an "administratively down" state.

If you are preparing for an OIR of a SPA, it is not necessary to independently shut down each of the interfaces prior to deactivation of the SPA. You do not need to independently restart any interfaces on a SPA after OIR of a SPA or SIP.

To shut down an interface on a SPA, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **shutdown**	Disables an interface.

To restart an interface on a SPA, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# **no shutdown**	Restarts a disabled interface.

Verifying the Interface Configuration

Besides using the **show running-configuration** command to display your Cisco 7600 series router configuration settings, you can use the **show interfaces pos** and **show controllers pos** commands to get detailed information on a per-port basis for your POS SPAs.

Verifying Per-Port Interface Status

The following example provides sample output for interface port 0 (the first port) on the SPA located in the subslot 0 of the SIP that is installed in slot 3 of the Cisco 7600 series router:

```
Router# show interfaces pos 3/0/0
POS3/0/0 is up, line protocol is up
  Hardware is Packet over Sonet
  MTU 4470 bytes, BW 622000 Kbit, DLY 100 usec,
     reliability 194/255, txload 1/255, rxload 1/255
  Encapsulation FRAME-RELAY, crc 16, loopback not set
  Keepalive set (10 sec)
  Scramble disabled
  LMI enq sent  18, LMI stat recvd 0, LMI upd recvd 0
  LMI enq recvd 1473, LMI stat sent  1473, LMI upd sent  0, DCE LMI up
  LMI DLCI 1023  LMI type is CISCO  frame relay DCE
  FR SVC disabled, LAPF state down
  Broadcast queue 0/256, broadcasts sent/dropped 2223/1, interface
broadcasts 1977
  Last input 00:00:05, output 00:00:05, output hang never
  Last clearing of "show interface" counters 04:46:02
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     47019 packets input, 163195100 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
```

```
 14332 runts, 925 giants, 0 throttles
          0 parity
 17820 input errors, 1268 CRC, 0 frame, 0 overrun, 0 ignored, 10 abort
 49252 packets output, 170900767 bytes, 0 underruns
 0 output errors, 0 applique, 2 interface resets
 0 output buffer failures, 0 output buffers swapped out
 3 carrier transitions.
```

Monitoring Per-Port Interface Statistics

The following is sample output from the **show controllers pos** command on a Cisco 7600 series router for POS interface 4/3/0 (which is the interface for port 0 of the SPA in subslot 3 of the SIP in chassis slot 4):

```
Router# show controllers pos 4/3/0
POS4/3/0
SECTION
  LOF = 0          LOS    = 0                            BIP(B1) = 65535
LINE
  AIS = 0          RDI    = 0          FEBE = 65535      BIP(B2) = 16777215
PATH
  AIS = 0          RDI    = 0          FEBE = 65535      BIP(B3) = 65535
  PLM = 0          UNEQ   = 0          TIM  = 0          TIU     = 0
  LOP = 0          NEWPTR = 3          PSE  = 0          NSE     = 0
Active Defects: None
Active Alarms:  None
Alarm reporting enabled for: SF SLOS SLOF B1-TCA B2-TCA PLOP B3-TCA

Framing: SONET
APS

  COAPS = 1          PSBF = 0
  State: PSBF_state = False
  Rx(K1/K2): 00/00  Tx(K1/K2): 00/00
  Rx Synchronization Status S1 = 00
  S1S0 = 00, C2 = CF
  Remote aps status (none); Reflected local aps status (none)
CLOCK RECOVERY
  RDOOL = 0
  State: RDOOL_state = False
PATH TRACE BUFFER: STABLE
  Remote hostname : woodson
  Remote interface: POS3/0/0
  Remote IP addr  : 0.0.0.0
  Remote Rx(K1/K2): 00/00  Tx(K1/K2): 00/00

BER thresholds:  SF = 10e-3  SD = 10e-6
TCA thresholds:  B1 = 10e-6  B2 = 10e-6  B3 = 10e-6

  Clock source:  internal
```

Configuration Examples

This section includes the following examples for configuring a POS SPA installed in a Cisco 7600 series router:

- Basic Interface Configuration Example, page 15-17
- MTU Configuration Example, page 15-17

Basic Interface Configuration Example

The following example shows how to enter global configuration mode to enter global configuration mode to specify the interface that you want to configure, configure an IP address for the interface, enable the interface, and save the configuration. This example configures interface port 0 (the first port) of the SPA located in subslot 0 of the SIP that is installed in slot 2 of the Cisco 7600 series router:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Specify the interface address
!
Router(config)# interface pos 2/0/0
!
! Configure an IP address
!
Router(config-if)# ip address 192.168.50.1 192.255.255.0
!
! Enable the interface
!
Router(config-if)# no shutdown
!
! Save the configuration to NVRAM
!
Router(config-if)# exit
Router# copy running-config startup-config
```

MTU Configuration Example

The following example sets the MTU to 4470 bytes on interface port 1 (the second port) of the SPA located in the bottom subslot (1) of the SIP that is installed in slot 2 of the Cisco 7600 series router:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Specify the interface address
!
```

```
Router(config)# interface pos 2/1/1
!
! Configure MTU
!
Router(config-if)# mtu 4470
```

POS Framing Configuration Example

The following example shows how to change from the default POS framing of SONET to SDH:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Specify the interface address
!
Router(config)# interface pos 2/1/1
! (The default pos framing is sonet)
!
!Modify the framing type
!
Router(config-if)# pos framing sdh
```

Keepalive Configuration Example

The following example shows how to change from the default keepalive period of 10 seconds to 20 seconds:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Specify the interface address
!
Router(config)# interface pos 2/1/1
!
! Configure keepalive 20
!
Router(config-if)# keepalive 20
```

CRC Configuration Example

The following example shows how to change the CRC size from 32 bits to the default 16 bits for POS SPAs:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Specify the interface address
!
Router(config)# interface pos 2/1/1
!
! Configure crc 16
!
Router(config-if)# crc 16
```

Clock Source Configuration Example

The following example shows how to change from the default clock source of internal to line:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Specify the interface address
!
Router(config)# interface pos 2/1/1
!
! Configure the clock source
!
Router(config-if)# clock source line
```

SONET Payload Scrambling Configuration Example

The following example shows how to change from a default SONET payload scrambling of disabled to enabled:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Specify the interface address
!
Router(config)# interface pos 2/1/1
!
! Configure the SONET payload scrambling
!
Router(config-if)# pos scramble-atm
```

Encapsulation Configuration Example

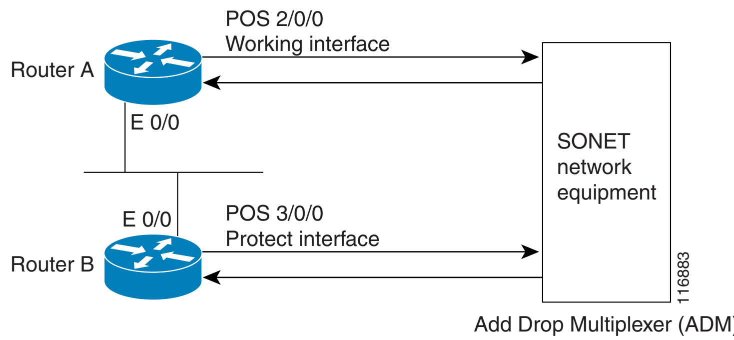
The following example shows how to change from the default encapsulation method of HDLC to PPP:

```
!Enter global configuration mode
!
Router# configure terminal

! Specify the interface address

Router(config)# interface pos 2/1/1
!
! Configure ppp
!
Router(config-if)# encapsulation ppp
```

APS Configuration Example

The following example shows the configuration of APS on router A and router B, and how to configure more than one protect or working interface on a router by using the **aps group** command. See Figure 15-1.

Figure 15-1 Basic APS Configuration



In this example, router A is configured with the working interface and router B is configured with the protect interface. If the working interface on router A becomes unavailable, the connection will automatically switch over to the protect interface on router B. The loopback interface is used as the interconnect. The **aps group** command is used even when a single protect group is configured.

The following example shows how to configure Router A for this scenario:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Configure a loopback interface as the protect interconnect path
!
Router(config)# interface loopback 1
Router(config-if)# ip address 10.10.10.10 255.0.0.0

! Configure the POS interface address for the APS working interface
!
Router(config)# interface pos 2/0/0
!
! Configure the POS interface IP address and other interface parameters
!
Router(config-if)# ip address 172.16.1.8 255.255.0.0
Router(config-if)# no ip directed-broadcast
Router(config-if)# no keepalive
Router(config-if)# crc 32
!
! Configure the APS group number by which to associate APS interfaces
!
Router(config-if)# aps group 1
!
! Configure a circuit number for the APS working interface
!
Router(config-if)# aps working 1
```

The following example shows how to configure Router B for this scenario:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Configure the POS interface address for the APS protect interface
!
Router(config)# interface pos 3/0/0
!
! Configure the POS interface IP address and other interface parameters
!
```

```
Router(config-if)# ip address 172.16.1.9 255.255.0.0
Router(config-if)# no ip directed-broadcast
Router(config-if)# no keepalive
Router(config-if)# crc 32
!
! Configure the APS group number by which to associate APS interfaces
!
Router(config-if)# aps group 1
!
! Configure a circuit number for the protect interface and an IP address for the router
! that has the APS working interface. In this case, the loopback interface address is
! used.
!
Router(config-if)# aps protect 1 10.10.10.10
```

POS Alarm Trigger Delays Configuration Example

The following example shows how to change POS line-level and path-level alarm trigger delays from the default of 100 milliseconds to 200 milliseconds:

```
!Enter global configuration mode
!
Router# configure terminal
!
! Specify the interface address
!
Router(config)# interface pos 2/1/1
!
Router(config-if)# pos delay triggers line 200
Router(config-if)# pos delay triggers path 200
```

SDCC Configuration Example

```
Router(config-if)# exit
Router(config))# hw-module subslot 1/0 srp mate 1/1
!
! Configure an SRP interface
!
Router(config)# interface srp 1/0/0
Router(config-if)# mac-address 0003.0003.0003
Router(config-if)# ip address 10.4.4.1 255.255.255.0
Router(config-if)# no ip directed-broadcast
Router(config-if)# ipv6 address 10:4:4::1/64
Router(config-if)# service-policy output parent
```

Part 7

Serial Shared Port Adapters

CHAPTER 16

Overview of the Serial SPAs

This chapter provides an overview of the release history, features, and supported MIBs for the Cisco 7600 SIP-200 with the 2- and 4-Port T3/E3 SPAs, the 8-Port Channelized T1/E1 SPA, the 1-Port Channelized OC-3/STM-1 SPA, and the -2 or 4-Port CT3 SPA.

This chapter includes the following sections:

Release History

Release	Modification
Cisco IOS Release 12.2(33)SRC	Support for the following software features was introduced on the Cisco 7600 SIP-200 on the Cisco 7600 series router: • Programmable BERT pattern enhancements for the 1-Port Channelized OC-3/STM-1 SPA and the 2- and 4-Port CT3 SPAs
Cisco IOS Release 12.2(33)SRA	Support for the following hardware was introduced on the Cisco 7600 SIP-200 on the Cisco 7600 series router: • 1-Port Channelized OC-3/STM-1 SPA
Cisco IOS Release 12.2(18)SXE	Support for the following hardware was introduced on the Cisco 7600 SIP-200 on the Cisco 7600 series router and Catalyst 6500 series switch: • 2-Port T3/E3 SPA (SPA-2XT3/E3) • 4-Port T3/E3 SPA (SPA-4XT3/E3) • 8-Port T1/E1 SPA (SPA-8XCHT1/E1) • 2-Port CT3 SPA (SPA-2XCT3/DS0) • 4-Port CT3 SPA (SPA-4XCT3/DS0)

Supported Features

This section provides a list of some of the primary features supported by the Cisco 7600 SIP-200 and SPA hardware and software.

- Online insertion and removal (OIR).
- Supports up to four single-height or two double-height Shared Port Adaptors (SPAs).
- Field Programmable Gate Array (FPGA) upgrade support.
- The SIP-200 supports the standard FPGA upgrade methods for the Cisco 7600 series router.

Restrictions

This section provides a list of Cisco 7600 SIP-200 configuration restrictions.

Note For other SIP-specific features and restrictions see also Chapter 3, "Overview of the SIPs and SSC" in this guide.

- On a 2-port or 4-port Channelized T3 SPA, when one of the T3 ports is configured as DS3 clear channel interface and the other T3s are configured with large number (greater than or equal to 400) of low bandwidth channels (NxDS0, N=1, 2, 3, or 4), the DS3 clear channel interface is not able to run at 100% DS3 line rate when those low bandwidth channels are idle (that is, not transmitting or receiving packets). This issue does not occur if those low bandwidth channels are not idle.
- On a 2-Port and 4-Port Channelized T3 SPA or 1-Port Channelized OC-3/STM-1 SPA, the maximum number of channels is limited to 1023 per SPA.
- On a 2-Port and 4-Port Channelized T3 SPA or 1-Port Channelized OC-3/STM-1 SPA, the maximum number of FIFO buffers is 4096. The FIFO buffers are shared among the interfaces; how they are shared is determined by speed. If all the FIFO buffers have been assigned to existing interfaces, a new interface cannot be created, and the "%Insufficient FIFOs to create channel group" error message is seen. FIFO allocation information is provided in Table 16-1.

  To find the number of available FIFO buffers, use the **show controller t3** command:

```
Router# show controller t3 3/0/0

T3 3/0/0 is up.
  Hardware is SPA-4XCT3/DS0
  IO FPGA version: 2.6, HDLC Framer version: 0
  T3/T1 Framer(1) version: 2, T3/T1 Framer(2) version: 2
  SUBRATE FPGA version: 1.4
  HDLC controller available FIFO buffers 3112
```

Table 16-1 FIFO Allocation

Number of Timeslots	Number of FIFO Buffers
1-6 DS0	4
7-8 DS0	6
9 DS0	6
10-12 DS0	8

Table 16-1 FIFO Allocation (continued)

Number of Timeslots	Number of FIFO Buffers
13–23 DS0	12
1–6 E1 TS	4
7–9 E1 TS	6
11–16 E1 TS	8
17–31 E1 TS	16
T1	12
E1	16
DS3	336

SPA Features

The following is a list of some of the significant software features supported by the 2- and 4-Port T3/E3 SPA, the 8-Port Channelized T1/E1 SPA, the 1-Port Channelized OC-3/STM-1 SPA, and the 2- and 4-Port CT3 SPAs.

- Software selectable between T1, E1, T3 or E3 framing on each card (ports are configured as all T1, E1, T3, or E3). Applies to the 2- and 4-Port T3/E3 SPA and 8-Port Channelized T1/E1 SPA.
- Layer 2 encapsulation support:
  - Point-to-Point Protocol (PPP)
  - High-level Data Link Control (HDLC)
  - Frame Relay
- Internal or network clock (selectable per port)
- Online insertion and removal (OIR)
- Hot standby router protocol (HSRP)
- Alarm reporting: 24-hour history maintained, 15-minute intervals on all errors
- 16- and 32-bit cyclic redundancy checks (CRC) supported (16-bit default)
- Local and remote loopback
- Bit error rate testing (BERT) pattern generation and detection per port

Note BERT is not supported on the 8-Port Channelized T1/E1 SPA.

- Programmable BERT patterns enhancements

Note The programmable BERT patterns enhancements are not supported on the 2- and 4-Port T3/E3 SPAs or the 8-Port Channelized T1/E1 SPA.

- Dynamic provisioning—Dynamic provisioning allows for the addition of new customer circuits within a channelized interface without affecting other customers.
- FPD (field programmable device upgrades)

- End-to-end FRF.12 fragmentation support
- Link Fragmentation and Interleaving (LFI) support
- Compressed Real-Time Protocol (cRTP)—8-Port Channelized T1/E1 SPA and 2-Port and 4-Port Channelized T3 SPA only. For more information about configuring cRTP, see the "Configuring Compressed Real-Time Protocol" section on page 4-4.
- T1 features
  - All ports can be fully channelized down to DS0
  - Data rates in multiples of 56 Kbps or 64 Kbps per channel
  - Maximum 1.536 Mbps for each T1 port
  - D4 Superframe (SF) and Extended Superframe (ESF) support for each T1 port
  - ANSI T1.403 and AT&T TR54016 CI FDL support
  - Internal and receiver recovered clocking modes
  - Short haul and long haul channel service unit (CSU) support
  - Binary eight-zero substitution (B8ZS) and alternate mark inversion (AMI) line encoding

Note B8ZS and AMI line encoding are not configurable for TW on the 2-Port and 4-Port Channelized T3 SPA.

  - Support for Multilink Point to Point Protocol (MLPPP) for full T1s on the same SPA (hardware based) and across SPAs (software based)
  - Support for Multilink Frame Relay (MLFR)
- E1 features
  - Maximum 1.984 Mbps for each E1 port in framed mode and a 2.048 Mbps in unframed E1 mode
  - All ports can be fully channelized down to DS0
  - Compliant with ITU G7.03, G.704, ETSI and ETS300156
  - Internal and receiver recovered clocking modes
  - Hi-density bipolar with three zones (HDB3) and AMI line encoding
  - Support for MLPPP for full E1s on the same SPA (hardware based) and across SPAs (software based).
  - Support for MLFR
- E3 features
  - Full duplex connectivity at E3 rate (34.368 MHz)
  - Supports ITU-T G.751 or G.832 framing (software selectable)
  - HD3B line coding
  - Compliant with E3 pulse mask
  - Line build-out: configured for up to 450 feet (135 m) of type 728A or equivalent coaxial cable
  - Loopback modes: data terminal equipment (DTE), local, dual, and network
  - E3 alarm/event detection (once per second polling)

    - Alarm indication signal (AIS)

    - Loss of frame (LOF)
    - Remote alarm indication (RAI)
  - Subrate and scrambling features for these data service unit (DSU) vendors:
    - Digital Link
    - ADC Kentrox
- T3 features
  - Binary 3-zero substitution (B3ZS) line coding
  - Compliant with DS3 pulse mask per ANSI T1.102-1993
  - DS3 far-end alarm and control (FEAC) channel support
  - Full duplex connectivity at DS3 rate (44.736 MHz)
  - 672 DS0s per T3
  - Loopback modes: DTE, local, remote, dual, and network
  - C-bit or M23 framing (software selectable)
  - Line build-out: configured for up to 450 feet (135 m) of type 734A or equivalent coaxial cable
  - DS3 alarm/event detection (once per second polling)
    - AIS
    - Out of frame (OOF)
    - Far-end receive failure (FERF)
  - Generation and termination of DS3 Maintenance Data Link (MDL) in C-bit framing
  - Full FDL support and FDL performance monitoring
  - Subrate and scrambling features for these DSU vendors:
    - Digital Link
    - ADC Kentrox
    - Adtran
    - Verilink
    - Larscom

Note On a 2-port or 4-port Channelized T3 SPA, when one of the T3 ports is configured as DS3 clear channel interface and the other T3s are configured with large number (greater than or equal to 400) of low bandwidth channels (NxDS0, N=1, 2, 3, or 4), the DS3 clear channel interface is not able to run at 100% DS3 line rate when those low bandwidth channels are idle (that is, not transmitting or receiving packets). This issue does not occur if those low bandwidth channels are not idle.

Supported MIBs

The following MIBs are supported in Cisco IOS Release 12.2S for the serial SPAs on the Cisco 7600 series router.

All serial SPAs:

- CISCO-ENTITY-ALARM-MIB

- CISCO-CLASS-BASED-QOS-MIB
- CISCO-ENVMON-MIB (For NPEs, NSEs, line cards, and SIPs only)
- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- CISCO-ENTITY-SENSOR-MIB
- ENTITY-MIB
- IF-MIB
- RMON-MIB
- MPLS-LDP-MIB
- MPLS-LSR-MIB
- MPLS-TE-MIB
- MPLS-VPN-MIB

2- and 4-Port T3/E3 SPAs:

- DS3/E3 MIB

8-Port Channelized T1/E1 SPA:

- DS1/E1 MIB

2- or 4-Port CT3 SPA:

- DS1-MIB
- DS3-MIB
- CISCO-FRAME-RELAY-MIB
- IANAifType-MIB
- RFC1381-MIB

For more information about MIB support on the Cisco 7600 series router, refer to the *Cisco 7600 Series Internet Router MIB Specifications Guide* found at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/7600mibs/index.htm

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

Displaying the SPA Hardware Type

To verify the SPA hardware type that is installed in your Cisco 7600 series router, you can use the **show diagbus** command or the **show interface** command (once the interface has been configured). There are several other commands on the Cisco 7600 series router that also provide SPA hardware information.

Table 16-2 shows the hardware description that appears in the **show** command output for each type of SPA that is supported on the Cisco 7600 series router.

Table 16-2 SPA Hardware Descriptions in show Commands

SPA	Description in show interfaces and show controllers Commands
4-Port T3/E3 SPA	"Hardware is SPA-4XT3/E3"
2-Port T3/E3 SPA	"Hardware is SPA-2XT3/E3"
8-Port Channelized T1/E1 SPA	"Hardware is SPA-T1E1"
2-Port CT3 SPA	"Hardware is 2 ports CT3 SPA"
4-Port CT3 SPA	"Hardware is 4 ports CT3 SPA"

Examples of the show interface Command

The following example shows output from the **show interface serial 5/0/0** command on a Cisco 7600 series router with a 4-Port T3/E3 SPA installed in slot 5:

```
Serial5/0/0 is up, line protocol is up
Hardware is SPA-4XT3/E3[3/0]
MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
reliability 248/255, txload 1/255, rxload 1/255
Encapsulation HDLC, crc 16, loopback not set
Keepalive set (10 sec)
Last input 00:00:06, output 00:00:07, output hang never
Last clearing of ''show interface'' counters 00:00:01
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts (0 IP multicast)
0 runts, 0 giants, 0 throttles
0 parity
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 applique, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
```

The following example shows output from the **show interface serial 6/0/1** command on a Cisco 7600 series router with a 8-Port Channelized T1/E1 SPA installed in slot 6:

```
Serial6/0/1:0 is up, line protocol is up
  Hardware is SPA-T1E1
  MTU 1500 bytes, BW 1536 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, crc 16, loopback not set
  Keepalive set (10 sec)
```

```
  LCP Open, multilink Open
  Last input 00:00:03, output 00:00:03, output hang never
  Last clearing of "show interface" counters 5d17h
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 3194905708
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
     74223 packets input, 1187584 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     74227 packets output, 1187751 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     0 output buffer failures, 0 output buffers swapped out
     4 carrier transitions no alarm present
  Timeslot(s) Used:1-24, subrate: 64Kb/s, transmit delay is 0 flags
```

Examples of the show controllers Command

The following example shows output from the **show controller serial** command on a Cisco 7600 series router with a 4-Port T3/E3 SPA installed in slot 5:

```
Router# show controllers serial 5/0/2
Serial5/0/2 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 0, since reset 0
   Data in current interval (807 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 306 Unavailable Secs
     500 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     564 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 2:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     564 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs

[output omitted]
```

The following example shows output from the **show controllers** command on a Cisco 7600 series router with a 8-Port Channelized T1/E1 SPA installed in slot 6:

```
Router# show controllers t1
T1 6/0/0 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
  alarm-trigger is not set
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (394 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
```

```
      0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
   Total Data (last 24 hours)
      0 Line Code Violations, 0 Path Code Violations,
      0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins,
      0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
T1 6/0/1 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
  alarm-trigger is not set
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (395 seconds elapsed):
      0 Line Code Violations, 0 Path Code Violations
      0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
      0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
  Total Data (last 24 hours)
      0 Line Code Violations, 0 Path Code Violations,
      0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins,
      0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
```

The following example shows output from the **show controllers** command on a Cisco 7600 series router with a 4-Port CT3 SPA installed in slot 3:

```
Router# show controllers t3
T3 3/1/2 is up.  Hardware is 4 ports CT3 SPA
  ATLAS FPGA version: 0, FREEDM336 version: 0
  TEMUX84(1) version: 0, TEMUX84(1) version: 0
  SUBRATE FPGA version: 0
  Applique type is Channelized T3
  No alarms detected.
  Framing is M23, Line Code is B3ZS, Clock Source is Internal
  Equipment customer loopback
  Data in current interval (146 seconds elapsed):
      0 Line Code Violations, 0 P-bit Coding Violation
      0 C-bit Coding Violation, 0 P-bit Err Secs
      0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
      0 Unavailable Secs, 0 Line Errored Secs
      0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
      0 Severely Errored Line Secs
      0 Far-End Errored Secs, 0 Far-End Severely Errored Secs
      0 CP-bit Far-end Unavailable Secs
      0 Near-end path failures, 0 Far-end path failures
      0 Far-end code violations, 0 FERF Defect Secs
      0 AIS Defect Secs, 0 LOS Defect Secs

  T1 1 is up
  timeslots: 1-24
  FDL per AT&T 54016 spec.
  No alarms detected.
  Framing is ESF, Clock Source is Internal
  Data in current interval (104 seconds elapsed):
      0 Line Code Violations, 0 Path Code Violations
      0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
      0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs
      0 Unavail Secs, 0 Stuffed Secs
      0 Near-end path failures, 0 Far-end path failures, 0 SEF/AIS Secs
  Total Data (last 2 15 minute intervals):
      0 Line Code Violations,0 Path Code Violations,
      0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins,
      0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs
      0 Unavail Secs, 0 Stuffed Secs
      0 Near-end path failures, 0 Far-end path failures, 0 SEF/AIS Secs
```

CHAPTER 17

Configuring the 8-Port Channelized T1/E1 SPA

This chapter provides information about configuring the 8-Port Channelized T1/E1 SPA on the Cisco 7600 series router. It includes the following sections:

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Configuration Tasks

This section describes how to configure the 8-Port Channelized T1/E1 SPA for the Cisco 7600 series router and includes information about verifying the configuration.

It includes the following topics:

Required Configuration Tasks

This section lists the required configuration steps to configure the 8-Port Channelized T1/E1 SPA. Some of the required configuration commands implement default values that might be appropriate for your network. If the default value is correct for your network, then you do not need to configure the command.

- Setting the Card Type
- Enabling the Interfaces on the Controller
- Verifying Controller Configuration

- Setting the IP Address
- Verifying Interface Configuration

Note To better understand the address format used to specify the physical location of the SIP, SPA, and interfaces, see the "Specifying the Interface Address on a SPA" section on page 17-6.

Setting the Card Type

The SPA is not functional until the card type is set. Information about the SPA is not indicated in the output of any **show** commands until the card type has been set. There is no default card type.

Note Mixing of interface types is not supported. All ports on a SPA must be of the same type.

To set the card type for the 8-Port Channelized T1/E1 SPA, complete these steps:

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **card type** {**e1** | **t1**} *slot subslot*	Sets the serial mode for the SPA: • **t1**—Specifies T1 connectivity of 1.536 Mbps. B8ZS is the default line code for T1. • **e1**—Specifies a wide-area digital transmission scheme used predominantly in Europe that carries data at a rate of 1.984 Mbps in framed mode and a 2.048 Mbps in unframed E1 mode. • *slot subslot*—Specifies the location of the SPA. See the "Specifying the Interface Address on a SPA" section on page 17-6.
Step 3	Router(config)# **exit**	Exits configuration mode and returns to the EXEC command interpreter prompt.

Enabling the Interfaces on the Controller

To create the interfaces for the 8-Port Channelized T1/E1 SPA, complete these steps:

	Command	Purpose
Step 1	Router(config)# **controller** {**t1** | **e1**} *slot/subslot/port*	Select the controller to configure and enter controller configuration mode. • **t1**—Specifies the T1 controller. • **e1**—Specifies the E1 controller. • *slot/subslot/port*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 17-6.
Step 2	Router(config-controller)# **clock source** {**internal** | **line**}	Sets the clock source. **Note** The clock source is set to internal if the opposite end of the connection is set to line and the clock source is set to line if the opposite end of the connection is set to internal. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used. This is the default for T1 and E1.
Step 3	Router(config-controller)# **linecode** {**ami** | **b8zs** | **hdb3**}	Selects the linecode type. • **ami**—Specifies Alternate Mark Inversion (AMI) as the linecode type. Valid for T1 and E1 controllers. • **b8zs**—Specifies binary 8-zero substitution (B8ZS) as the linecode type. Valid for T1 controller only. This is the default for T1 lines. • **hdb3**—Specifies high-density binary 3 (hdb3) as the linecode type. Valid for E1 controller only. This is the default for E1 lines.
Step 4	**For T1 controllers** Router(config-controller)# **framing** {**sf** | **esf**} **For E1 controllers** Router(config-controller)# **framing** {**crc4** | **no-crc4**}	Selects the framing type. • **sf**—Specifies Super Frame as the T1 frame type. • **esf**—Specifies Extended Super Frame as the T1 frame type. This is the default for E1. • **crc4**—Specifies CRC4 as the E1 frame type. This is the default for E1. • **no-crc4**—Specifies no CRC4 as the E1 frame type.

	Command	Purpose
Step 5	Router(config-controller)# **channel-group** *t1 t1-number* {**timeslots** *range* | **unframed**} [**speed** {**56** | **64**}]	Define the time slots that belong to each T1 or E1 circuit. • *t1 t1-number*— Channel-group number. When configuring a T1 data line, channel-group numbers can be values from 1 to 28. When configuring an E1 data line, channel-group numbers can be values from 0 to 30. • **timeslots** *range*— One or more time slots or ranges of time slots belonging to the channel group. The first time slot is numbered 1. For a T1 controller, the time slot range is from 1 to 24. For an E1 controller, the time slot range is from 1 to 31. • **unframed**—Unframed mode (G.703) uses all 32 time slots for data. None of the 32 time slots are used for framing signals. • **speed**—(Optional) Speed of the underlying DS0s. – **56**— – **64**— **Note** The default is 64 is speed is not mentioned in the config. **Note** Each channel group is presented to the system as a serial interface that can be configured individually. **Note** Once a channel group has been created with the **channel-group** command, the channel group cannot be changed without removing the channel group. To remove a channel group, see the section Changing a Channel Group Configuration, page 17-17.
Step 6	Router(config)# **exit**	Exits configuration mode and returns to the EXEC command interpreter prompt.

Verifying Controller Configuration

Use the **show controllers** command to verify the controller configuration:

```
Router(config)# show controllers t1
T1 6/0/1 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
  alarm-trigger is not set
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (395 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
  Total Data (last 24 hours)
```

```
     0 Line Code Violations, 0 Path Code Violations,
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins,
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
```

Setting the IP Address

To set the IP address for the 8-Port Channelized T1/E1 SPA, complete these steps:

	Command	Purpose
Step 1	Router(config)# **interface serial** *slot/subslot/port***:***channel-group*	Selects the interface to configure from global configuration mode. • *slot/subslot/port:channel-group*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 17-6.
Step 2	Router(config-if)# **ip address** *address mask*	Sets the IP address and subnet mask. • *address*—IP address. • *mask*—Subnet mask.
Step 3	Router(config)# **exit**	Exits configuration mode and returns to the EXEC command interpreter prompt.

Verifying Interface Configuration

Use the **show interfaces** command to verify the interface configuration:

```
Router(config)# show interfaces
.
.
.
Serial6/0/1:0 is up, line protocol is up
  Hardware is SPA-T1E1
  MTU 1500 bytes, BW 1536 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, crc 16, loopback not set
  Keepalive set (10 sec)
  LCP Open, multilink Open
  Last input 00:00:03, output 00:00:03, output hang never
  Last clearing of "show interface" counters 5d17h
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 3194905708
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
     74223 packets input, 1187584 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     74227 packets output, 1187751 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     0 output buffer failures, 0 output buffers swapped out
     4 carrier transitions no alarm present
  Timeslot(s) Used:1-24, subrate: 64Kb/s, transmit delay is 0 flags
.
.
```

Specifying the Interface Address on a SPA

SPA interface ports begin numbering with "0" from left to right. Single-port SPAs use only the port number 0. To configure or monitor SPA interfaces, you need to specify the physical location of the SIP, SPA, and interface in the CLI. The interface address format is *slot*/*subslot*/*port*, where:

- *slot*—Specifies the chassis slot number in the Cisco 7600 series router where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA.

The following example shows how to specify the first interface (0) on a SPA installed in the first subslot of a SIP (0) installed in chassis slot 3:

```
Router(config)# interface serial 3/0/0
```

This command shows a serial SPA as a representative example, however the same *slot*/*subslot*/*port* format is similarly used for other SPAs (such as ATM and POS) and other non-channelized SPAs.

For the 8-Port Channelized T1/E1 SPA, the interface address format is *slot/subslot/port*:*channel-group*, where:

- *channel-group*—Specifies the logical channel group assigned to the timeslots within the T1 link.

For more information about identifying slots and subslots, see the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section on page 4-2.

Optional Configurations

There are several standard, but optional, configurations that might be necessary to complete the configuration of your serial SPA.

Configuring Framing

Framing is used to synchronize data transmission on the line. Framing allows the hardware to determine when each packet starts and ends. To configure framing, use the following commands.

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **controller {t1 | e1}** *slot/subslot/port*	Selects the controller to configure. • **t1**—Specifies the T1 controller. • **e1**—Specifies the E1 controller. • *slot/subslot/port*—Specifies the location of the controller. See the "Specifying the Interface Address on a SPA" section on page 17-6.
For T1 controllers Router(config-controller)# **framing {sf | esf}** **For E1 controllers** Router(config-controller)# **framing {crc4 | no-crc4}**	Set the framing on the interface. • **sf**—Specifies Super Frame as the T1 frame type. • **esf**—Specifies extended Super Frame as the T1 frame type. This is the default. for T1. • **crc4**—Specifies CRC4 frame as the E1 frame type. This is the default for E1. • **no-crc4**—Specifies no CRC4 frame as the E1 frame type.

Verifying Framing Configuration

Use the **show controllers** command to verify the framing configuration:

```
Router# show controllers t1
T1 6/0/0 is down.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  Receiver has loss of frame.
  alarm-trigger is not set
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (717 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 717 Unavail Secs
  Total Data (last 24 hours)
     0 Line Code Violations, 0 Path Code Violations,
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins,
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 86400 Unavail Secs
```

Configuring Encapsulation

When traffic crosses a WAN link, the connection needs a Layer 2 protocol to encapsulate traffic. To set the encapsulation method, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port***:***channel-group*	Selects the interface to configure. • *slot/subslot/port:channel-group*—Specifies the location of the interface. See: "Specifying the Interface Address on a SPA" section on page 17-6
Router(config-if)# **encapsulation** *encapsulation-type* **{hdlc | ppp | frame-relay}**	Set the encapsulation method on the interface. • **hdlc**—High-Level Data Link Control (HDLC) protocol for serial interface. This encapsulation method provides the synchronous framing and error detection functions of HDLC without windowing or retransmission. This is the default for synchronous serial interfaces. • **ppp**—PPP (for serial interface). • **frame-relay**—Frame Relay (for serial interface).

Verifying Encapsulation

Use the **show interfaces serial** command to verify encapsulation on the interface:

```
Router# show interfaces serial 6/0/0:0
Serial6/0/0:0 is down, line protocol is down
  Hardware is SPA-T1E1
  MTU 1500 bytes, BW 1536 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, crc 32, loopback not set
  Keepalive set (10 sec)
  LCP Closed, multilink Closed
  Last input 1w0d, output 1w0d, output hang never
  Last clearing of "show interface" counters 6d23h
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 1152 kilobits/sec
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
```

```
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions alarm present
  Timeslot(s) Used:1-24, subrate: 64Kb/s, transmit delay is 0 flags
```

Configuring the CRC Size for T1

All 8-Port Channelized T1/E1 SPA interfaces use a 16-bit cyclic redundancy check (CRC) by default, but also support a 32-bit CRC. CRC is an error-checking technique that uses a calculated numeric value to detect errors in transmitted data. The designators 16 and 32 indicate the length (in bits) of the frame check sequence (FCS). A CRC of 32 bits provides more powerful error detection, but adds overhead. Both the sender and receiver must use the same setting.

CRC-16, the most widely used CRC throughout the United States and Europe, is used extensively with WANs. CRC-32 is specified by IEEE 802 and as an option by some point-to-point transmission standards. It is often used on Switched Multimegabit Data Service (SMDS) networks and LANs.

To set the length of the cyclic redundancy check (CRC) on a T1 interface, use these commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port***:***channel-group*	Selects the interface to configure. • *slot/subslot/port:channel-group*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 17-6.
Router(config-if)# **crc** {**16** | **32**}	Selects the CRC size in bits. • **16**—16-bit CRC. This is the default • **32**—32-bit CRC.

Verifying the CRC Size

Use the **show interfaces serial** command to verify the CRC size set on the interface:

```
Router# show interfaces serial 6/0/0:0
Serial6/0/0:0 is up, line protocol is up
  Hardware is SPA-T1E1
  MTU 1500 bytes, BW 1536 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, crc 32, loopback not set
  Keepalive set (10 sec)
  LCP Open, multilink Open
  Last input 00:00:38, output 00:00:00, output hang never
  Last clearing of "show interface" counters 01:46:16
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
     1272 packets input, 20396 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     6 input errors, 3 CRC, 0 frame, 0 overrun, 0 ignored, 3 abort
     1276 packets output, 20460 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

```
    0 carrier transitions no alarm present
 Timeslot(s) Used:1-24, subrate: 64Kb/s, transmit delay is 0 flags
```

Configuring FDL

Facility Data Link (FDL) is a 4-kbps channel provided by the Extended Super Frame (ESF) T1 framing format. The FDL performs outside the payload capacity and allows you to check error statistics on terminating equipment without intrusion.

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **controller t1** *slot/subslot/port*	Selects the controller to configure. • *slot/subslot/port*—Specifies the location of the controller. See the "Specifying the Interface Address on a SPA" section on page 17-6.
Router(config-controller)# **fdl** [**ansi** | **att** | **both**]	If the framing format was configured for **esf**, configures the format used for Facility Data Link (FDL). • **ansi**—Select ansi for FDL to use the ANSI T1.403 standard. • **att**—Select att for FDL to use the AT&T TR54016 standard. • **both—**Specifies support for both AT&T technical reference 54016 and ANSI T1.403 for ESF FDL exchange support.

Verifying FDL

Use the **show controllers t1** command to verify the **fdl** setting:

```
Router# show controllers t1
T1 6/0/1 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
  alarm-trigger is not set
  Framing is ESF, FDL is ansi, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (742 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
  Total Data (last 73 15 minute intervals):
     1278491 Line Code Violations, 3 Path Code Violations,
     0 Slip Secs, 1 Fr Loss Secs, 177 Line Err Secs, 0 Degraded Mins,
     3 Errored Secs, 0 Bursty Err Secs, 1 Severely Err Secs, 227 Unavail Secs
.
.
.
```

Configuring Multilink Point-to-Point Protocol (Hardware-based)

Multilink Point to Point Protocol (MLPPP) allows you to combine T1 or E1 lines into a bundle that has the combined bandwidth of multiple T1/E1 lines. You choose the number of bundles and the number of T1 or E1 lines in each bundle.

MLPPP for T1/E1 Configuration Guidelines

The required conditions are:

- Only T1 or E1 links in a bundle
- All links on the same SPA
- Maximum of 12 links in a bundle.

Note Some notes about hardware-based MLPPP:

Only 3 fragmentation sizes are possible 128, 256 and 512 bytes

Fragmentation is enabled by default, default size is 512 bytes

Fragmentation size is configured using the **ppp multilink fragment-delay** command after using the **interface multilink** command. The least of the fragmentation sizes (among the 3 sizes possible) satisfying the delay criteria is configured. (For example, a 192 byte packet causes a delay of 1 millisecond on a T1 link, so the nearest fragmentation size is 128 bytes.)

The **show ppp multilink** command indicates the MLPPP type and the fragmentation size:

```
Router# show ppp multilink
Multilink1, bundle name is Patriot2
Bundle up for 00:00:13
Bundle is Distributed
0 lost fragments, 0 reordered, 0 unassigned
0 discarded, 0 lost received, 206/255 load
0x0 received sequence, 0x0 sent sequence
Member links: 2 active, 0 inactive (max not set, min not set)
Se4/2/0/1:0, since 00:00:13, no frags rcvd
Se4/2/0/2:0, since 00:00:10, no frags rcvd
Distributed fragmentation on. Fragment size 512.  Multilink in Hardware.
```

Fragmentation is disabled explicitly by using the **no ppp multilink fragmentation** command after using the **interface multilink** command.

Create a Multilink Bundle

To create a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** *group-number*	Creates a multilink interface and enters multilink interface mode. • *group-number*—The group number for the multilink bundle.
Router(config-if)# **ip address** *address mask*	Sets the IP address for the multilink group. • *address*—The IP address. • *mask*—The IP netmask.

Assign an interface to a Multilink Bundle

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port/t1-number***:***channel-group*	Selects the interface to configure and enters interface configuration mode. See the "Specifying the Interface Address on a SPA" section on page 17-6. • *slot/subslot/port/t1-number***:***channel-group*—Selects the interface to configure.
Router(config-if)# **encapsulation ppp**	Enables PPP encapsulation.
Router(config-if)# **multilink-group** *group-number*	Assigns the interface to a multilink bundle. • *group-numbe*r—The multilink group number for the T1 or E1 bundle.
Router(config-if)# **ppp multilink**	Enables multilink PPP on the interface.
Repeat these commands for each interface you want to assign to the multilink bundle.	

Configuring fragmentation size on an MLPPP Bundle (optional)

To configure the fragmentation size on a multilink PPP bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** *slot/subslot/port/t1-number***:***channel-group*	Creates a multilink interface and enters multilink interface mode. • *channel-group*—The group number for the multilink bundle. Range 1 to 2147483647.
Router(config-if)# **ppp multilink fragment-delay** *delay*	Sets the fragmentation size satisfying the configured delay on the multilink bundle. • *delay*—delay in milliseconds

Disabling the fragmentation on an MLPPP Bundle (optional)

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** *group-number*	Creates a multilink interface and enters multilink interface mode. • *group-number*—The group number for the multilink bundle. Range 1 to 2147483647.
Router(config-if)# **no ppp multilink fragmentation**	Disables the fragmentation on the multilink bundle.

Verifying Multilink PPP

Use the **show ppp multilink** command to verify the PPP multilinks:

```
Router# show ppp multilink
Multilink1, bundle name is mybundle
    Bundle up for 01:40:50
    Bundle is Distributed
    0 lost fragments, 0 reordered, 0 unassigned
    0 discarded, 0 lost received, 1/255 load
    0x0 received sequence, 0x0 sent sequence
Member links: 5 active, 0 inactive (max not set, min not set)
    Se6/0/0/1:0, since 01:40:50, no frags rcvd
    Se6/0/1/1:0, since 01:40:09, no frags rcvd
    Se6/0/3/1:0, since 01:15:44, no frags rcvd
    Se6/0/4/1:0, since 01:03:17, no frags rcvd
    Se6/0/6/1:0, since 01:01:06, no frags rcvd
    Se6/0/6:0, since 01:01:06, no frags rcvd
```

Configuring MLFR for T1/E1

Multilink Frame Relay (MLFR) allows you to combine T1/E1 lines into a bundle that has the combined bandwidth of multiple T1/E1 lines. You choose the number of bundles and the number of T1/E1 lines in each bundle. This allows you to increase the bandwidth of your network links beyond that of a single T1/E1 line.

MLFR for T1/E1 Configuration Guidelines

MLFR will function in hardware if all of the following conditions are met:

- Only T1 or E1 member links
- All links are on the same SPA
- Maximum of 12 links in a bundle

Create a Multilink Bundle

To create a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface mfr** *number*	Configures a multilink Frame Relay bundle interface. • *number*—The number for the Frame Relay bundle.
Router(config-if)# **frame-relay multilink bid** *name*	(Optional) Assigns a bundle identification name to a multilink Frame Relay bundle. • *name*—The name for the Frame Relay bundle. **Note** The bundle identification (BID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shut** and **no shut** commands in interface configuration mode.

Assign an Interface to a Multilink Bundle

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port***:***channel-group*	Selects the interface to assign. • *slot/subslot/port:channel-group*—Specifies the location of the interface. See the "Specifying the Interface Address on a SPA" section on page 17-6.

Command	Purpose
Router(config-if)# **encapsulation frame-relay mfr** *number* [*name*]	Creates a multilink Frame Relay bundle link and associates the link with a bundle. • *number*—The number for the Frame Relay bundle. • *name*—The name for the Frame Relay bundle.
Router(config-if)# **frame-relay multilink lid** *name*	(Optional) Assigns a bundle link identification name with a multilink Frame Relay bundle link. • *name*—The name for the Frame Relay bundle. **Note** The bundle link identification (LID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shut** and **no shut** commands in interface configuration mode.
Router(config-if)# **frame-relay multilink hello** *seconds*	(Optional) Configures the interval at which a bundle link will send out hello messages. The default value is 10 seconds. • *seconds*—Number of seconds between hello messages sent out over the multilink bundle.
Router(config-if)# **frame-relay multilink ack** *seconds*	(Optional) Configures the number of seconds that a bundle link will wait for a hello message acknowledgment before resending the hello message. The default value is 4 seconds. • *seconds*—Number of seconds a bundle link will wait for a hello message acknowledgment before resending the hello message.
Router(config-if)# **frame-relay multilink retry** *number*	(Optional) Configures the maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment. The default value is 2 tries. • *number*—Maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment.

Verifying Multilink Frame Relay

Use the **show frame-relay multilink detailed** command to verify the Frame Relay multilinks:

```
router# show frame-relay multilink detailed

Bundle: MFR49, State = down, class = A, fragmentation disabled
 BID = MFR49
 No. of bundle links = 1, Peer's bundle-id =
 Bundle links:

  Serial6/0/0:0, HW state = up, link state = Add_sent, LID = test
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
```

```
    Peer LID = , RTT = 0 ms
    Statistics:
    Add_link sent = 21, Add_link rcv'd = 0,
    Add_link ack sent = 0, Add_link ack rcv'd = 0,
    Add_link rej sent = 0, Add_link rej rcv'd = 0,
    Remove_link sent = 0, Remove_link rcv'd = 0,
    Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
    Hello sent = 0, Hello rcv'd = 0,
    Hello_ack sent = 0, Hello_ack rcv'd = 0,
    outgoing pak dropped = 0, incoming pak dropped = 0
```

Invert Data on the T1/E1 Interface

If the interface on the 8-Port Channelized T1/E1 SPA is used to drive a dedicated T1 line that does not have B8ZS encoding, you must invert the data stream on the connecting CSU/DSU or on the interface. Be careful not to invert data on both the CSU/DSU and the interface, as two data inversions will cancel each other out. To invert data on a T1/E1 interface, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port***:***channel-group*	Selects the serial interface.
Router(config-if)# **invert data**	Inverts the data stream.

Use the **show running configuration** command to verify that invert data has been set:

```
Router# show running configuration
.
.
.
interface Serial6/0/0:0
 no ip address
 encapsulation ppp
 logging event link-status
 load-interval 30
 invert data
 no cdp enable
 ppp chap hostname group1
 ppp multilink
 multilink-group 1
!
.
.
.
```

Changing a Channel Group Configuration

To alter the configuration of an existing channel group, the channel group needs to be removed first. To remove an existing channel group, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **controller** {**t1** | **e1**} *slot/subslot/port*	Select the controller to configure and enter controller configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See: Specifying the Interface Address on a SPA, page 17-6.
Router(config-controller)# **no channel-group** *t1 t1-number*	Select the channel group you want to remove. • *t1 t1-number*—Channel-group number.
Follow the steps in the section: Enabling the Interfaces on the Controller, page 17-3.	Create a new channel group with the new configuration.

Configuring Multipoint Bridging

Multipoint bridging (MPB) enables the connection of multiple ATM PVCs, Frame Relay PVCs, BCP ports, and WAN Gigabit Ethernet subinterfaces into a single broadcast domain (virtual LAN), together with the LAN ports on that VLAN. This enables service providers to add support for Ethernet-based Layer 2 services to the proven technology of their existing ATM and Frame Relay legacy networks. Customers can then use their current VLAN-based networks over the ATM or Frame Relay cloud. This also allows service providers to gradually update their core networks to the latest Gigabit Ethernet optical technologies, while still supporting their existing customer base.

For MPB configuration guidelines and restrictions and feature compatibility tables, see the "Configuring Multipoint Bridging" section on page 4-23.

Configuring Bridging Control Protocol Support

The Bridging Control Protocol (BCP) enables forwarding of Ethernet frames over SONET networks and provides a high-speed extension of enterprise LAN backbone traffic through a metropolitan area. The implementation of BCP on the SPAs includes support for IEEE 802.1D, IEEE 802.1Q Virtual LAN (VLAN), and high-speed switched LANs.

For BCP configuration guidelines and restrictions and feature compatibility tables, see the "Configuring PPP Bridging Control Protocol Support" section on page 4-35.

Configuring BCP on MLPPP

BCP on MLPPP Configuration Guidelines

- Only Distributed MLPPP is supported
- Only channelized interfaces allowed, and member links must be from the same controller card
- Only trunk port BCP is supported on MLPPP
- Bridging can be configured only on the bundle interface

Note BCP on MLPPP operates only in trunk mode. For more inforation on trunk mode, see the "Configuring BCP in Trunk Mode" section on page 4-38.

Configuring BCP on MLPPP Trunk Mode

To configure BCP on MLPPP trunk mode, perform these steps:

	Command	Purpose
Step 1	Router(config)# **interface multilink**	Selects the multilink interface.
Step 2	Router(config-if)# **switchport**	Puts an interface that is in Layer 3 mode into Layer 2 mode for Layer 2 configuration.
Step 3	Router(config-if)# **switchport trunk allowed vlan 100**	By default, no VLANs are allowed. Use this command to explicitly allow VLANs; valid values for *vlan-list* are from 1 to 4094.
Step 4	Router(config-if)# **switchport mode trunk**	Configures the router port connected to the switch as a VLAN trunk port.
Step 5	Router(config-if)# **switchport nonegotiate**	Puts the LAN port into permanent trunking mode but prevents the port from generating DTP frames
Step 6	Router(config-if)# **no ip address**	Removes the assigned IP address.
Step 7	Router(config-if)# **ppp multilink**	Enables this interface to support MLP.
Step 8	Router(config-if)# **multilink-group 1**	Assigns this interface to the multilink group.
Step 9	Router(config-if)# **interface Serial1/0/0.1/1/1/1:0**	Designates a serial interface as a multilink bundle.
Step 10	Router(config-if)# **no ip address**	Unassigns the IP address.
Step 11	Router(config-if)# **encapsulation ppp**	Enables PPP encapsulation.
Step 12	Router(config-if)# **ppp multilink**	Enables this interface to support MLP.
Step 13	Router(config-if)# **multilink-group 1**	Assigns this interface to the multilink group 1.
Step 14	Router(config-if)# **interface Serial1/0/0.1/1/1/2:0**	Designates a serial interface as a multilink bundle.
Step 15	Router(config-if)# **no ip address**	Unassigns the IP address.
Step 16	Router(config-if)# **encapsulation ppp**	Enables PPP encapsulation.
Step 17	Router(config-if)# **ppp multilink**	Enables this interface to support MLP.
Step 18	Router(config-if)# **multilink-group 1**	Assigns this interface to the multilink group 2.
Step 19	Router(config-if)# **shutdown**	Shuts down an interface.
Step 20	Router(config-if)# **no shutdown**	Reopens an interface.
Step 21	Router(config-if)# **switchport trunk allowed vlan** vlan-list	By default, no VLANs are allowed. Use this command to explicitly allow VLANs; valid values for *vlan-list* are from 1 to 4094.

Verifying BCP on MLPPP Trunk Mode

To display information about Multilink PPP, use the **show ppp multilink** command in EXEC mode.

Command	Purpose
Router(config-if)# **show ppp multilink**	Displays information on a multilink group.

The following shows an example of **show ppp multilink**:

```
Router# show ppp multilink

Multilink1, bundle name is group 1
Bundle is Distributed
0 lost fragments, 0 reordered, 0 unassigned, sequence 0x0/0x0 rcvd/sent
0 discarded, 0 lost received, 1/255 load
Member links: 4 active, 0 inactive (max no set, min not set)
Serial1/0/0/:1
Serial1/0/0/:2
Serial1/0/0/:3
Serial1/0/0/:4
```

FRF.12 Guidelines

FRF.12 functions in hardware. Note the following:

- The fragmentation is configured at the main interface
- Only 3 fragmentation sizes are available: 128 bytes, 256 bytes, and 512 bytes.

LFI Guidelines

LFI can function two ways—using FRF.12 or MLPPP. MLPPP LFI can be done in both hardware and software while FRF.12 LFI is done only in hardware.

HW MLPPP LFI Guidelines

LFI using MLPPP will function only in hardware if there is just one member link in the MLPPP bundle. The link can be a fractional T1 or full T1. Note the following:

- The **ppp multilink interleave** command needs to be configured to enable interleaving.
- Only three fragmentation sizes are supported: 128 bytes, 256 bytes, and 512 bytes.
- Fragmentation is enabled by default, the default size being 512 bytes.
- A policy-map having a priority class needs to applied to main interface.
- When hardware-based LFI is enabled, fragmentation counters are not displayed.

FRF.12 LFI Guidelines

LFI using FRF.12 is always done is hardware. Note the following:

- The fragmentation is configured at the main interface
- Only 3 fragmentation sizes are available: 128 bytes, 256 bytes, and 512 bytes.
- A policy-map having a priority class needs to applied to main interface.

Configuring QoS Features on Serial SPAs

The SIPs and SPAs support many QoS features using modular QoS CLI (MQC) configuration. For information about the QoS features supported by the serial SPAs, see the "Configuring QoS Features on a SIP" section on page 4-61.

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), use the following command in privileged EXEC configuration mode:

Command	Purpose
Router# **copy running-config startup-config**	Writes the new configuration to NVRAM.

For more information about managing configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

Verifying the Interface Configuration

Besides using the **show running-configuration** command to display your Cisco 7600 series router configuration settings, you can use the **show interfaces serial** and the **show controllers serial** commands to get detailed information on a per-port basis for your 8-Port Channelized T1/E1 SPA.

Verifying Per-Port Interface Status

To find detailed interface information on a per-port basis for the 8-Port Channelized T1/E1 SPA, use the **show interfaces serial** command.

The following example provides sample output for interface port 0 on the SPA located in the first subslot of the SIP installed in slot 6 of a Cisco 7609 router:

```
Router# show interface serial 6/0/0:0
Serial6/0/0:0 is up, line protocol is up
  Hardware is SPA-T1E1
  MTU 1500 bytes, BW 1536 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, crc 32, loopback not set
  Keepalive set (10 sec)
  LCP Open, multilink Open
  Last input 00:00:38, output 00:00:00, output hang never
```

```
  Last clearing of "show interface" counters 01:46:16
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
     1272 packets input, 20396 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     6 input errors, 3 CRC, 0 frame, 0 overrun, 0 ignored, 3 abort
     1276 packets output, 20460 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions no alarm present
  Timeslot(s) Used:1-24, subrate: 64Kb/s, transmit delay is 0 flags
```

Configuration Examples

This section includes the following configuration examples:

- Framing and Encapsulation Configuration Example, page 17-21
- CRC Configuration Example, page 17-22
- Facility Data Link Configuration Example, page 17-22
- MLPPP Configuration Example, page 17-22
- Invert Data on the T1/E1 Interface Example, page 17-24
- MFR Configuration Example, page 17-23

Framing and Encapsulation Configuration Example

The following example sets the framing and encapsulation for the controller and interface:

```
! Specify the controller and enter controller configuration mode
!
Router(config)# controller t1 6/0/0
!
! Specify the framing method
!
Router(config-controller)# framing esf
!
! Exit controller configuration mode and return to global configuration mode
!
Router(config-controller)# exit
!
! Specify the interface and enter interface configuration mode
!
Router(config)# interface serial 6/0/0:0
!
! Specify the encapsulation protocol
!
Router(config-if)# encapsulation ppp
!
! Exit interface configuratin mode
!
Router(config-if)# exit
```

```
!
! Exit global configuration mode
!
Router(config)# exit
```

CRC Configuration Example

The following example sets the CRC size for the interface:

```
! Specify the interface and enter interface configuration mode
!
Router(config)# interface serial 6/0/0:0
!
! Specify the CRC size
!
Router(config-if)# crc 32
!
! Exit interface configuration mode and return to global configuration mode
!
Router(config-if)# exit
!
! Exit global configuration mode
!
Router(config)# exit
```

Facility Data Link Configuration Example

The following example configures Facility Data Link:

```
! Specify the controller and enter controller configuration mode
!
Router(config)# controller t1 6/0/0
!
! Specify the FDL specification
!
Router(config-controller)# fdl ansi
!
! Exit controller configuration mode and return to global configuration mode
!
Router(config-controller)# exit
!
! Exit global configuration mode
!
Router(config)# exit
```

MLPPP Configuration Example

The following example creates a PPP Multilink bundle:

```
! Enter global configuration mode
!
Router# configure terminal
!
! Create a multilink bundle and assign a group number to the bundle
!
Router(config)# interface multilink 1
!
! Specify an IP address for the multilink group
```

```
!
Router(config-if)# ip addres 123.456.789.111 255.255.255.0
!
! Enable Multilink PPP
!
Router(config-if)# ppp multilink
!
! Leave interface multilink configuration mode
!
Router(config-if)# exit
!
! Specify the interface to assign to the multilink bundle
!
Router(config)# interface serial 3/1//0:1
!
! Enable PPP encapsulation on the interface
!
Router(config-if)# encapsulation PPP
!
! Assign the interface to a multilink bundle
!
Router(config-if)# multilink-group 1
!
! Enable Multilink PPP
!
Router(config-if)# ppp multilink
!
! Exit interface configuration mode
!
Router(config-if)# exit
!
! Exit global configuration mode
!
Router(config)# exit
```

MFR Configuration Example

The following example configures Multilink Frame Relay (MFR):

```
! Create a MFR interface and enter interface configuration mode
!
Router(config)# interface mfr 49
!
! Assign the bundle identification (BID) name 'test' to a multilink bundle.
!
Router(config-if)# frame-relay multilink bid test
!
! Exit interface configuration mode and return to global configuration mode
!
Router(config-if)# exit
!
! Specify the serial interface to assign to a multilink bundle
!
Router(config)# interface serial 5/1/3:0
!
! Creates a multilink Frame Relay bundle link and associates the link with a multilink
bundle
!
Router(config-if)# encapsulation frame-relay mfr 49
!
! Assigns a bundle link identification (LID) name with a multilink bundle link
!
```

```
Router(config-if)# frame-relay multilink lid test
!
! Configures the interval at which the interface will send out hello messages
!
Router(config-if)# frame-relay multilink hello 15
!
! Configures the number of seconds the interface will wait for a hello message
acknowledgement before resending the hello message
!
Router(config-if)# frame-relay multilink ack 6
!
! Configures the maximum number of times the interface will resend a hello message while
waiting for an acknowledgement
!
Router(config-if)# frame-relay multilink retry 5
!
! Exit interface configuration mode and return to global configuration mode
!
Router(config-if)# exit
!
! Exit global configuration mode
!
Router(config)# exit
```

Invert Data on the T1/E1 Interface Example

The following example inverts the data on the serial interface:

```
! Enter global configuration mode
!
Router# configure terminal
!
! Specify the serial interface and enter interface configuration mode
!
Router(config)# interface serial 5/1/3:0
!
! Configure invert data
!
Router(config-if)# invert data
!
! Exit interface configuration mode and return to global configuration mode
!
Router(config-if)# exit
!
! Exit global configuration mode
!
Router(config)# exit
```

CHAPTER 18

Configuring the 2-Port and 4-Port Clear Channel T3/E3 SPAs

This chapter provides information about configuring the 2-Port and 4-Port Clear Channel T3/E3 Shared Port Adapters (SPAs) on the Cisco 7600 series router. It includes the following sections:

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Configuration Tasks

This section describes how to configure the 2-Port Clear Channel T3/E3 SPA for the Cisco 7600 series router and includes information about verifying the configuration.

It includes the following topics:

Required Configuration Tasks

This section lists the required configuration steps to configure the 2-Port and 4-Port Clear Channel T3/E3 SPA. Some of the required configuration commands implement default values that might be appropriate for your network. If the default value is correct for your network, then you do not need to configure the command.

- Setting the Card Type
- Configure the Interface

Note To better understand the address format used to specify the physical location of the Spa Interface Processor (SIP), SPA, and interfaces, see the: "Specifying the Interface Address on a SPA" section on page 18-5.

Setting the Card Type

The SPA is not functional until the card type is set. Information about the SPA is not indicated in the output of any show commands until the card type has been set. There is no default card type.

Note Mixing of interface types is not supported. All ports on a SPA will be the of the same type.

To set the card type for the 2-Port and 4-Port Clear Channel T3/E3 SPA, complete these steps:

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **card type** {**t3** | **e3**} *slot subslot*	Sets the serial mode for the SPA: • **t3**—Specifies T3 connectivity of 44210 kbps through the network, using B3ZS coding. • **e3**—Specifies a wide-area digital transmission scheme used predominantly in Europe that carries data at a rate of 34010 kbps. • *slot subslot*—Specifies the location of the SPA. See the: "Specifying the Interface Address on a SPA" section on page 18-5
Step 3	Router(config)# **exit**	Exit configuration mode and return to the EXEC command interpreter prompt.

Configure the Interface

To set the ip address for the 2-Port and 4-Port Clear Channel T3/E3 SPA, complete these steps:

	Command	Purpose
Step 1	Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure and enters interface configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5
Step 2	Router(config-if)# **ip address** *address mask*	Sets the IP address and subnet mask. • *address*—IP address • *mask*—Subnet mask
Step 3	Router(config-if)# **clock source** {**internal** | **line**}	Sets the clock source to internal. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used. This is the default.
Step 4	Router(config-if)# **no shut**	Enables the interface.
Step 5	Router(config)# **exit**	Exits configuration mode and returns to the EXEC command interpreter prompt.

Verifying Controller Configuration

Use the **show controllers** command to verify the controller configuration:

```
Router# show controllers serial 6/0/0
Serial6/0/0 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 2, since reset 0
   Data in current interval (546 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
.
.
.
Data in Interval 44:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     560 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
   Total Data (last 44 15 minute intervals):
     0 Line Code Violations, 0 P-bit Coding Violation,
```

```
     0 C-bit Coding Violation,
     0 P-bit Err Secs, 0 P-bit Sev Err Secs,
     0 Sev Err Framing Secs, 0 Unavailable Secs,
     24750 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs

   Transmitter is sending AIS.

   Receiver has loss of signal.

    40434 Sev Err Line Secs, 0 Far-End Err Secs, 0 Far-End Sev Err Secs
    0 P-bit Unavailable Secs, 0 CP-bit Unavailable Secs
    0 CP-bit Far-end Unavailable Secs
    0 Near-end path failures, 0 Far-end path failures

   No FEAC code is being received
  MDL transmission is disabled
```

Use the **show controllers brief** command to view a subset of the **show controllers** output:

```
Router# show controllers serial 6/0/2 brief
Serial6/0/2 -
   Framing is c-bit, Clock Source is Internal
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 0, since reset 22

   No alarms detected.

   No FEAC code is being received
  MDL transmission is disabled
```

Verifying Interface Configuration

Use the **show interfaces** command to verify the interface configuration:

```
Router# show interfaces serial 6/0/0
Serial6/0/0 is up, line protocol is up
  Hardware is SPA-4T3E3
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 12/255, rxload 56/255
  Encapsulation FRAME-RELAY, crc 16, loopback not set
  Keepalive set (10 sec)
  LMI enq sent  13477, LMI stat recvd 13424, LMI upd recvd 0, DTE LMI up
  LMI enq recvd 19, LMI stat sent  0, LMI upd sent  0
  LMI DLCI 1023  LMI type is CISCO  frame relay DTE
  FR SVC disabled, LAPF state down
  Broadcast queue 0/256, broadcasts sent/dropped 0/0, interface broadcasts 0
  Last input 00:00:09, output 00:00:09, output hang never
  Last clearing of "show interface" counters 1d13h
  Input queue: 0/75/3/3891 (size/max/drops/flushes); Total output drops: 5140348
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 9716000 bits/sec, 28149 packets/sec
  5 minute output rate 2121000 bits/sec, 4466 packets/sec
     14675957334 packets input, 645694448563 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     14562482078 packets output, 640892196653 bytes, 0 underruns
     0 output errors, 0 applique, 4 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
   rxLOS inactive, rxLOF inactive, rxAIS inactive
   txAIS inactive, rxRAI inactive, txRAI inactive
```

```
Serial6/0/0.16 is up, line protocol is up
  Hardware is SPA-4T3E3
  Internet address is 110.1.1.2/24
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 11/255, rxload 53/255
  Encapsulation FRAME-RELAY
```

Specifying the Interface Address on a SPA

SPA interface ports begin numbering with "0" from left to right. Single-port SPAs use only the port number 0. To configure or monitor SPA interfaces, you need to specify the physical location of the SIP, SPA, and interface in the CLI. The interface address format is *slot*/*subslot*/*port*, where:

- *slot*—Specifies the chassis slot number in the Cisco 7600 series router where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA.

The following example shows how to specify the first interface (0) on a SPA installed in the first subslot of a SIP (0) installed in chassis slot 3:

```
Router(config)# interface serial 3/0/0
```

This command shows a serial SPA as a representative example, however the same *slot*/*subslot*/*port* format is similarly used for other SPAs (such as ATM and POS) and other non-channelized SPAs.

For more information about identifying slots and subslots, see the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section on page 4-2.

Optional Configurations

There are several standard, but optional configurations that might be necessary to complete the configuration of your serial SPA.

Configuring Data Service Unit Mode

Configure the SPA to connect with customer premise Data Service Units (DSUs) by setting the DSU mode. Subrating a T3 or E3 interface reduces the peak access rate by limiting the data transfer rate. To configure the DSU mode and bandwidth, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** s*lot/subslot/port*	Selects the interface to configure and enters interface configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5
T3 Router(config-if)# **dsu mode** {**0** | **1** | **2** | **3** | **4**} **E3** Router(config-if)# **dsu mode** {**0** | **1**}	Specifies the interoperability mode used by a T3 controller. • **0**—Connects a T3/E3 controller to another T3/E3 controller or to a Digital Link DSU (DL3100 in T3 mode and DL3100E in E3 mode). This is the default. • **1**—Connects a T3/E3 controller to a Kentrox DataSMART T3/E3 IDSU. • **2**—Connects a T3 controller to a Larscom Access-T45 DS3 DSU. • **3**—Connects a T3 controller to an Adtran T3SU 300. • **4**—Connects a T3 controller to a Verilink HDM 2182.

Command	Purpose
Router(config-if)# **dsu bandwidth** *kbps*	Specifies the allowable bandwidth. • *kbps*—The bandwidth range and increment values are based on the specific DSU. Default for T3 mode is 44010 kbps and 34010 kbps for E3 mode. • Digital Link DL3100 – range: 300 to 44210 kbps – increments: 300 kbps • Digital Link DL3100E – range: 358 to 34010 kbps – increments: 358 kbps • Kentrox DataSMART T3/E3 IDSU – range: 1000 to 34000 kbps (E3 mode) – range: 1500 to 44210 kbps (T3 mode) – increments: 500 kbps • Larscom Access-T45 DS3 – range: 3100 to 44210 kbps – increments: 3100 kbps • Adtran T3SU 300 – range: 80 to 44210 kbps – increments: 80 kbps • Verilink HDM 2182 – range: 1600 to 31600 kbps – increments: 1600 kbps
Router(config-if)# **remote** {**accept** | **fullrate**}	Specifies where the DSU bandwidth is set. • **accept**—Accept incoming remote requests to reset the DSU bandwidth. • **fullrate**—Set far end DSU to its fullrate bandwidth.

Verifying DSU Mode

Use the **show controllers serial** command to display the DSU settings:

```
Router# show controllers serial 6/0/0
Serial6/0/0 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 2, since reset 0
   Data in current interval (546 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
```

```
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
.
.
.
```

Configuring Maintenance Data Link

MDL messages are used to communicate identification information between local and remote ports. The type of information included in MDL messages includes the equipment identification code (EIC), location identification code (LIC), frame identification code (FIC), unit, Path Facility Identification (PFI), port number, and Generator Identification numbers.

Note C-bit framing has to be enabled in order to transport MDL messages between source and destination T3 ports.

To configure Maintenance Data Link (MDL), use the following commands.

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure. • *slot/subslot/port*—Specifies the location of the interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5

Command	Purpose
Router(config-if)# **mdl** [**string** {**eic** | **fic** | **generator** | **lic** | **pfi** | **port** | **unit**} *string*}] | [**transmit** {**idle-signal** | **path** | **test-signal**}]	Configures the Maintenance Data Link (MDL) message. • **eic** *string*—Equipment identification code (up to 10 characters), which is a value used to describe a specific piece of equipment according to ANSI T1.107-1995. • **fic** *string*—Frame identification code (up to 10 characters), which is a value used to identify where the equipment is located within a building at a given location according to ANSI T1.107-1995. • **generator** *string*—Specifies the Generator number string sent in the MDL Test Signal message; can be up to 38 characters. • **lic** *string*—Location identification code (up to 11 characters), which is a value used to describe a specific location according to ANSI T1.107-1995. • **pfi** *string*—Specifies the Path Facility Identification Code sent in the MDL Path message; can be up to 38 characters. • **port** *string*—Specifies the Port number string sent in the MDL Idle Signal message; can be up to 38 characters. • **unit** *string*—Unit identification code (up to 6 characters), which is a value that identifies the equipment location within a subslot according to ANSI T1.107-1995. • **transmit idle-signal**—Enables transmission of the MDL idle signal message. An MDL idle signal message, as defined by ANSI T1.107, is distinguished from path and test signal messages in that it contains a port number as its final data element. • **transmit path**—Enables transmission of the MDL path message. An MDL path message, as defined by ANSI T1.107, is distinguished from idle and test signal messages in that it contains a facility identification code as its final data element. • **transmit test-signal**—Enables transmission of the MDL test signal message. An MDL test signal message, as defined by ANSI T1.107, is distinguished from path and idle signal messages in that it contains a generator number as its final data element.

Verifying MDL

Use the **show controllers serial** command to display the MDL settings:

```
Router# show controllers serial 6/0/0
Serial6/0/0 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 2, since reset 0
   Data in current interval (546 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
.
.
.

  Data in Interval 96:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
   Total Data (last 24 hours)
     0 Line Code Violations, 0 P-bit Coding Violation,
     0 C-bit Coding Violation,
     0 P-bit Err Secs, 0 P-bit Sev Err Secs,
     0 Sev Err Framing Secs, 0 Unavailable Secs,
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs

   No alarms detected.

    0 Sev Err Line Secs, 1 Far-End Err Secs, 0 Far-End Sev Err Secs
    0 P-bit Unavailable Secs, 0 CP-bit Unavailable Secs
    0 CP-bit Far-end Unavailable Secs
    0 Near-end path failures, 0 Far-end path failures

No FEAC code is being received
  MDL transmission is enabled
     EIC: tst, LIC: 67,
     Test Signal GEN_NO: test
  Far-End MDL Information Received
     EIC: tst, LIC: 67,
     Test Signal GEN_NO: test
```

Configuring Scramble

T3/E3 scrambling is used to assist clock recovery on the receiving end. Scrambling is designed to randomize the pattern of 1s and 0s carried in the physical layer frame. Randomizing the digital bits can prevent continuous, nonvariable bit patterns—in other words, long strings of all 1s or all 0s. Several physical layer protocols rely on transitions between 1s and 0s to maintain clocking.

Scrambling can prevent some bit patterns from being mistakenly interpreted as alarms by switches placed between the Data Service Units (DSUs).

To configure scrambling, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure. • *slot/subslot/port*—Specifies the location of the interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5
Router(config-if)# [**no**] **scramble**	Enables scrambling. Scrambling is disabled by default. • **scramble**—Enable scramble. • **no scramble**—Disable scramble. **Note** When using framing bypass, **no scrambling** must be configured.

Verifying Scramble Configuration

Use the **show controllers serial** command to display the scrambling setting:

```
Router# show controllers serial 6/0/0
Serial6/0/0 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 2, since reset 0
   Scrambling is enabled
   Data in current interval (356 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
.
.
.
```

Configuring Framing

Framing is used to synchronize data transmission on the line. Framing allows the hardware to determine when each packet starts and ends. To configure framing, use the following commands.

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure. • *slot/subslot/port*—Specifies the location of the T3/E3 interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5
T3 Router(config-if)# **framing {bypass | c-bit | m13}** **E3** Router(config-if)# **framing {bypass | g751| g832}**	Sets the framing on the interface. • **bypass**—Configure framing bypass to use the full T3 or E3 bandwidth • **c-bit**—Specifies C-bit parity framing. This is the default for T3. • **m13**—Specifies M13 framing. • **g751**— Specifies g751 framing. This is the default for E3. • **g832**—Specifies g832 framing.

Verifying Framing Configuration

Use the **show controllers serial** command to display the framing method:

```
Router# show controllers serial 6/0/0
Serial6/0/0 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 2, since reset 0
   Data in current interval (546 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
.
.
.
```

Configuring Encapsulation

When traffic crosses a WAN link, the connection needs a Layer 2 protocol to encapsulate traffic. To set the encapsulation method, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure. • *slot/subslot/port*—Specifies the location of the interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5
Router(config-if)# **encapsulation** {**hdlc** | **ppp** | **frame-relay**}	Sets the encapsulation method on the interface. • **hdlc**—High-Level Data Link Control (HDLC) protocol for serial interface. This is the default. • **ppp**—PPP (for serial interface). • **frame-relay**—Frame Relay (for serial interface).

Verifying Encapsulation

Use the **show interfaces** command to display the encapsulation method:

```
Router# show interfaces serial 6/0/1
Serial6/0/1 is up, line protocol is up
  Hardware is SPA-4T3E3
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 223/255, rxload 222/255
  Encapsulation FRAME-RELAY, crc 16, loopback not set
  Keepalive set (10 sec)
  LMI enq sent  13076, LMI stat recvd 13076, LMI upd recvd 0, DTE LMI up
  LMI enq recvd 0, LMI stat sent  0, LMI upd sent  0
  LMI DLCI 0  LMI type is ANSI Annex D  frame relay DTE
  FR SVC disabled, LAPF state down
  Broadcast queue 0/256, broadcasts sent/dropped 0/0, interface broadcasts 0
  Last input 00:00:04, output 00:00:04, output hang never
  Last clearing of "show interface" counters 1d12h
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 38579000 bits/sec, 109611 packets/sec
  5 minute output rate 38671000 bits/sec, 109852 packets/sec
     14374551065 packets input, 632486376132 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     14408526130 packets output, 633974757440 bytes, 0 underruns
     0 output errors, 0 applique, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
   rxLOS inactive, rxLOF inactive, rxAIS inactive
   txAIS inactive, rxRAI inactive, txRAI inactive
```

Configuring Cable Length

The **cablelength** command compensates for the loss in decibels based on the distance from the device to the first repeater in the circuit. A longer distance from the device to the repeater requires that the signal strength on the circuit be boosted to compensate for loss over that distance. To configure cable length, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure and enters interface configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5
Router(config-if)# **cablelength** *length*	Sets the cable length. • *length*—Range is 0-450 feet. The default is 10 feet.

Verify Cable Length Setting

Use the **show interfaces serial** command to verify the cable length setting:

```
Router# show interfaces serial 4/0/0
Serial4/0/0 -
   Framing is c-bit, Clock Source is Internal
   Bandwidth limit is 44210, DSU mode 0, Cable length is 200
   rx FEBE since last clear counter 0, since reset 22
   Data in current interval (446 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 2:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
.
.
.
```

Configuring Invert Data

Delays between the TE clock and data transmission indicate that the transmit clock signal might not be appropriate for the interface rate and length of cable being used. Different ends of the wire may have variances that differ slightly. Invert the clock signal to compensate for these factors. To configure invert data, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure and enters interface configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5
Router(config-if)# **invert** {**data**}	Inverts the data. • **data—**Invert the data stream.

Verify Invert Data Setting

Use the **show running configuration** command to verify that invert data was set on the interface:

```
Router# show running configuration
.
.
.
interface Serial6/0/0
 ip address 51.1.1.1 255.255.255.0
 logging event link-status
 dsu bandwidth 44210
 framing c-bit
 cablelength 10
 clock source internal
 invert data
 mdl string eic tst
 mdl string lic 67
 mdl string generator test
 mdl transmit path
 mdl transmit test-signal
 no cdp enable
!
.
.
.
```

Configuring the Trace Trail Buffer

Configure TTB to send messages to the remote device. The TTB messages check for the continued presence of the transmitter. To configure TTB, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure and enters interface configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See the: "Specifying the Interface Address on a SPA" section on page 18-5
Router(config-if)# **ttb** {**country** | **rnode** | **serial** | **snode** | **soperator** | **x**} *string*	Sends a Trace Trail Buffer message in E3 g.832 framing mode. • **country**—Two character country code • **rnode**—Receive node code • **serial**—M.1400 serial • **snode**—Sending location/Node ID code • **soperator**—Sending operator code. (must be numeric) • **x**—X0 • *string*—TTB message.

Verify TTB Settings

Use the **show controllers serial** command to display the TTB settings for the interface:

```
Router# show controllers serial 6/0/0
Serial6/0/0 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 2, since reset 0
   Data in current interval (546 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
.
.
.
No alarms detected.
TTB transmission is disabled
TTB Rx: country: us soperator: s snode: sn rnode: rn x: x serial: 1
```

Configuring Multipoint Bridging

Multipoint bridging (MPB) enables the connection of multiple ATM PVCs, Frame Relay PVCs, BCP ports, and WAN Gigabit Ethernet subinterfaces into a single broadcast domain (virtual LAN), together with the LAN ports on that VLAN. This enables service providers to add support for Ethernet-based Layer 2 services to the proven technology of their existing ATM and Frame Relay legacy networks. Customers can then use their current VLAN-based networks over the ATM or Frame Relay cloud. This also allows service providers to gradually update their core networks to the latest Gigabit Ethernet optical technologies, while still supporting their existing customer base.

For MPB configuration guidelines and restrictions and feature compatibility tables, see the "Configuring Multipoint Bridging" section on page 4-23 of Chapter 4, "Configuring the SIPs and SSC."

Configuring Bridging Control Protocol Support

The Bridging Control Protocol (BCP) enables forwarding of Ethernet frames over SONET networks and provides a high-speed extension of enterprise LAN backbone traffic through a metropolitan area. The implementation of BCP on the SPAs includes support for IEEE 802.1D, IEEE 802.1Q Virtual LAN (VLAN), and high-speed switched LANs.

For BCP configuration guidelines and restrictions and feature compatibility tables, see the "Configuring PPP Bridging Control Protocol Support" section on page 4-35 of Chapter 4, "Configuring the SIPs and SSC."

Configuring QoS Features on Serial SPAs

The SIPs and SPAs support many QoS features using modular QoS CLI (MQC) configuration. For information about the QoS features supported by the serial SPAs, see the "Configuring QoS Features on a SIP" section on page 4-61 of Chapter 4, "Configuring the SIPs and SSC."

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), use the following command in privileged EXEC configuration mode:

Command	Purpose
Router# **copy running-config startup-config**	Writes the new configuration to NVRAM.

For more information about managing configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

Verifying the Interface Configuration

Besides using the **show running-configuration** command to display your Cisco 7600 series router configuration settings, you can use the **show interfaces serial** and the **show controllers serial** commands to get detailed information on a per-port basis for your 2-Port and 4-Port Clear Channel T3/E3 SPA.

Verifying Per-Port Interface Status

To find detailed interface information on a per-port basis for the 2-Port and 4-Port Clear Channel T3/E3 SPA, use the **show interfaces serial** command.

The following example provides sample output for interface port 1 on the SPA located in the first subslot of the SIP installed in slot 5 of a Cisco 7600 series router:

```
Router# show interface serial 5/0/1
Serial5/0/1 is up, line protocol is up
  Hardware is SPA-4T3E3
  Internet address is 120.1.1.1/24
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 234/255, rxload 234/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Last input 00:00:00, output 00:00:01, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 40685000 bits/sec, 115627 packets/sec
  5 minute output rate 40685000 bits/sec, 115624 packets/sec
     4652915554 packets input, 204728203496 bytes, 0 no buffer
     Received 4044 broadcasts (0 IP multicast)
     130 runts, 0 giants, 0 throttles
              0 parity
     1595 input errors, 543 CRC, 0 frame, 0 overrun, 0 ignored, 922 abort
     4653081242 packets output, 204735493748 bytes, 0 underruns
     0 output errors, 0 applique, 4 interface resets
     0 output buffer failures, 0 output buffers swapped out
     2 carrier transitions
```

Monitoring Per-Port Interface Statistics

To find detailed status and statistical information on a per-port basis for the 2-Port and 4-Port Clear Channel T3/E3 SPA, use the **show controllers serial** command.

The following example provides sample output for interface port 1 on the SPA located in the first subslot of the SIP that is installed in slot 5 of the Cisco 7600 series router:

```
show controller serial 5/0/2
Serial5/0/2 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 0, since reset 0
   Data in current interval (807 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 306 Unavailable Secs
     500 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     564 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 2:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
```

```
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     564 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 3:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     562 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 4:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     560 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
    .
    .
    .
    Total Data (last 44 15 minute intervals):
         0 Line Code Violations, 0 P-bit Coding Violation,
         0 C-bit Coding Violation,
         0 P-bit Err Secs, 0 P-bit Sev Err Secs,
         0 Sev Err Framing Secs, 0 Unavailable Secs,
         24750 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs

       Transmitter is sending AIS.

       Receiver has loss of signal.

        40434 Sev Err Line Secs, 0 Far-End Err Secs, 0 Far-End Sev Err Secs
        0 P-bit Unavailable Secs, 0 CP-bit Unavailable Secs
        0 CP-bit Far-end Unavailable Secs
        0 Near-end path failures, 0 Far-end path failures

       No FEAC code is being received
      MDL transmission is disabled
```

Configuration Examples

This section includes the following configuration examples:

DSU Configuration Example

The following example confgiures DSU on interface port 0 on slot 4, subslot 1.

```
! Specify the serial interface and enter interface configuration mode
!
Router(config)# interface serial 4/1/0
!
! Specify the DSU mode
!
Router(config-if)# dsu mode 0
!
! Specify the DSU bandwidth
!
Router(config-if)# dsu bandwidth 10000
!
! Set the DSU bandwidth to accept or reject the incoming remote requests
!
Router(config-if)# dsu remote accept
```

MDL Configuration Example

The following example configures the MDL strings on interface port 0 on slot 4, subslot 1.

```
! Specify the serial interface and enter interface configuration mode
!
Router(config)# interface serial 4/1/0
!
! Specify the MDL strings
!
Router(config-if)# mdl string eic beic
Router(config-if)# mdl string lic beic
Router(config-if)# mdl string fic bfix
Router(config-if)# mdl string unit bunit
Router(config-if)# mdl string pfi bpfi
Router(config-if)# mdl string port bport
Router(config-if)# mdl string generator bgen
Router(config-if)# mdl transmit path
Router(config-if)# mdl transmit idle-signal
Router(config-if)# mdl transmit test-signal
```

Scrambling Configuration Example

The following example configures scrambling on the T3/E3 interface:

```
! Enter global configuration mode
!
Router# configure terminal
!
! Specify the serial interface and enter interface configuration mode
!
Router(config)# interface serial 4/1/3
!
! Enable scrambling
!
Router(config-if)# scrambling
```

Framing Configuration Example

The following example configures framing on interface port 1 on slot 4, subslot 1.

```
! Specify the serial interface and enter interface configuration mode
```

```
!
Router(config)# interface serial 4/1/1
!
! Specify the framing method
!
Router(config-if)# framing m13
```

Encapsulation Configuration Example

The following example configures encapsulation on interface port 1 on slot 4, subslot 1.

```
! Specify the serial interface and enter interface configuration mode
!
Router(config)# interface serial 4/1/1
!
! Specify the encapsulation method
!
Router(config-if)# encapsulation PPP
```

Cable Length Configuration Example

The following example configures sets the cable length to 200 feet:

```
! Enter global configuration mode
!
Router# configure terminal
!
! Specify the serial interface and enter interface configuration mode
!
Router(config)# interface serial 4/1/3
!
! Specify the cable length
!
Router(config-if)# cablelength 200
```

Invert Data Configuration Example

The following example enables invert data:

```
! Enter global configuration mode
!
Router# configure terminal
!
! Specify the serial interface and enter interface configuration mode
!
Router(config)# interface serial 4/1/3
!
! Enable invert data
!
Router(config-if)# invert data
```

Trace Trail Buffer Configuration Example

The following example configures the TTB attributes:

```
! Enter global configuration mode
```

```
!
Router# configure terminal
!
! Specify the serial interface and enter interface configuration mode
!
Router(config)# interface serial 4/1/3
!
! Specify the TTB attributes
!
Router(config-if)# ttb country ab
Router(config-if)# ttb soperator 56
Router(config-if)# ttb snode 34
Router(config-if)# ttb rnode cd
Router(config-if)# ttb x 7
Router(config-if)# ttb serial 12
```

CHAPTER 19

Configuring the 2-Port and 4-Port Channelized T3 SPAs

This chapter provides information about configuring the 2-Port and 4-Port Channelized T3 Shared Port Adapters (SPAs) on the Cisco 7600 series router. It includes the following sections:

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Configuration Tasks

This section describes how to configure the serial SPAs for the Cisco 7600 series router and includes information about verifying the configuration.

It includes the following topics:

Required Configuration Tasks

This section lists the required configuration steps to configure the 2-Port and 4-Port Channelized T3 SPA. Some of the required configuration commands implement default values that might be appropriate for your network.

- Configuring the T3 Controller, page 19-2
- Configuring the Logical T1 Interfaces, page 19-3
- Verifying T3 Controller Configuration, page 19-5
- Verifying Interface Configuration, page 19-6

Note To better understand the address format used to specify the physical location of the SPA Interface Processor (SIP), SPA, and interfaces, see the section Specifying the Interface Address on a SPA, page 19-7.

Configuring the T3 Controller

To configure the T3 controller for the 2-Port and 4-Port Channelized T3 SPA, complete these steps:

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **controller t3** *slot/subslot/port*	Selects the controller to configure and enters controller configuration mode. • *slot/subslot/port*—Specifies the location of the CT3 SPA port. See: "Specifying the Interface Address on a SPA" section on page 19-7.
Step 3	Router(config-controller)# [**no**] **channelized**	(Optional) Specifies the channelization mode. • **channelized**—In channelized mode, the T3 link can be channelized into 28 T1s, and each T1 can be further channelized into 24 DS0s. This is the default. • **no channelized**—In the unchannelized mode the T3 link provides a single high-speed data channel of 44210 kbps.

	Command	Purpose
Step 4	Router(config-controller)# **framing** {**auto-detect** | **c-bit** | **m23**}	(Optional) Specifies the framing type in channelized mode. • **auto-detect**—Detects the framing type at the device at the end of the line and switches to that framing type. If both devices are set to auto-detect, c-bit framing is used. • **c-bit**—Specifies C-bit parity framing. This is the default. • **m23**—Specifies M23 framing. **Note** To set the framing type for an un-channelized T3, see: "Configuring T3 Framing" section on page 19-14.
Step 5	Router(config-controller)# **clock source** {**internal** | **line**}	(Optional) Specifies the clock source. • **internal**—Specifies that the internal clock source is used. Default for channelized mode. • **line**—Specifies that the network clock source is used. Default for un-channelized mode.
Step 6	Router(config-controller)# **cablelength** {**0** - **450**}	(Optional) Specifies the cable length. The default is 224 ft. • **0-450**—Cable length in feet.

Configuring the Logical T1 Interfaces

If channelized mode is configured for the T3 controller, use the following procedure to configure the logical T1 interfaces.

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **controller t3** *slot/subslot/port*	Selects the controller to configure and enters controller configuration mode. • *slot/subslot/port*—Specifies the location of the CT3 SPA port. See: "Specifying the Interface Address on a SPA" section on page 19-7

	Command	Purpose
Step 3	Router(config-controller)# **t1** *t1-number* **channel-group** *channel-number* **timeslots** *range* [**speed** {**56** | **64**}]	Specifies the T1 channel and timeslots to be mapped to each channel. • *t1-number*—T1 number from 1–28. • *channel-number*—Specifies a channel-group mapping(0–23) under the designated T1. • *range*—List of timeslots under the channel-group. Timeslots assigned to this T1 can be 1–24 or a combination of subranges within 1– 24. You can indicate a range using a hyphen, commas, or a combination of both. One timeslot equals one DS0. • **speed 56** or **64**— Specifies the speed of a timeslot as either 56 or 64 kbps. The default speed of 64 kbps is not mentioned in the config.
Step 4	Router(config-controller)# **t1** *t1-number* **framing** {**esf** | **sf** [**hdlc-idle** {**0x7e** | **0xff**}] [**mode** {**j1**}]}	(Optional) Specifies the T1 framing type using the **framing** command. • **sf**—Specifies Super Frame as the T1 frame type. **Note** If you select sf framing, you should consider disabling yellow alarm detection because the yellow alarm can be incorrectly detected with sf framing. • **esf**—Specifies Extended Super Frame as the T1 frame type. This is the default. • **hdlc-idle**— The hdlc-idle option allows you to set the idle pattern for the T1 interface to either **0x7e** (the default) or **0xff**.

	Command	Purpose
Step 5	Router(config-controller)# **t1** *channel-number* **clock source** {**internal** | **line**}	(Optional) Specifies the T1 clock source. • **internal**—Specifies that the internal clock source is used. This is the default. • **line**—Specifies that the network clock source is used.

Step 6 Configure the serial interfaces.

Note After a T1 channel is configured, it appears to the Cisco IOS software as a serial interface; therefore, all the configuration commands for a serial interface are available. However, not all commands are applicable to the T1 interface. All the encapsulation formats, such as PPP, HDLC, and Frame Relay are applicable to the configured T1. Encapsulation can be set via the serial interface configuration commands.

For detailed interface configuration information, see the *Cisco IOS Interface Configuration Guide, Release 12.2* at:
http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_book09186a0080087098.html

Verifying T3 Controller Configuration

Use the **show controllers** command to verify the controller configuration:

```
Router# show controllers t3
T3 3/1/0 is administratively down.
T3 3/1/1 is administratively down.
T3 3/1/2 is up.  Hardware is 4 ports CT3 SPA
  ATLAS FPGA version: 0, FREEDM336 version: 0
  TEMUX84(1) version: 0, TEMUX84(1) version: 0
  SUBRATE FPGA version: 0
  Applique type is Channelized T3
  No alarms detected.
  Framing is M23, Line Code is B3ZS, Clock Source is Internal
  Equipment customer loopback
  Data in current interval (746 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation, 0 P-bit Err Secs
     0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
     0 Unavailable Secs, 0 Line Errored Secs
     0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
     0 Severely Errored Line Secs
     0 Far-End Errored Secs, 0 Far-End Severely Errored Secs
     0 CP-bit Far-end Unavailable Secs
     0 Near-end path failures, 0 Far-end path failures
     0 Far-end code violations, 0 FERF Defect Secs
     0 AIS Defect Secs, 0 LOS Defect Secs

  T1 1 is up
  timeslots: 1-24
  FDL per AT&T 54016 spec.
  No alarms detected.
  Framing is ESF, Clock Source is Internal
  Data in current interval (177 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs
     0 Unavail Secs, 0 Stuffed Secs
     0 Near-end path failures, 0 Far-end path failures, 0 SEF/AIS Secs
```

```
  Total Data (last 2 15 minute intervals):
     0 Line Code Violations,0 Path Code Violations,
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins,
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs
     0 Unavail Secs, 0 Stuffed Secs
     0 Near-end path failures, 0 Far-end path failures, 0 SEF/AIS Secs

  T1 2
    Not configured.

  T1 3
    Not configured.
.
.
.
T3 3/1/3 is up.  Hardware is 4 ports CT3 SPA
  ATLAS FPGA version: 0, FREEDM336 version: 0
  TEMUX84(1) version: 0, TEMUX84(1) version: 0
  SUBRATE FPGA version: 0
  Applique type is Subrate T3
  No alarms detected.
  MDL transmission is disabled

  FEAC code received: No code is being received
  Framing is C-BIT Parity, Line Code is B3ZS, Clock Source is Line
  Equipment customer loopback
  Data in current interval (657 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation, 0 P-bit Err Secs
     0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
     0 Unavailable Secs, 0 Line Errored Secs
     0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
     0 Severely Errored Line Secs
     0 Far-End Errored Secs, 0 Far-End Severely Errored Secs
     0 CP-bit Far-end Unavailable Secs
     0 Near-end path failures, 0 Far-end path failures
     0 Far-end code violations, 0 FERF Defect Secs
     0 AIS Defect Secs, 0 LOS Defect Secs
```

Verifying Interface Configuration

Use the **show interface serial** command to verify the interface configuration. The following example shows the ouput for the serial interface for an un-channelized T3:

```
Router# show interface serial3/0/0
Serial3/0/0 is down, line protocol is down
  Hardware is Channelized/ClearChannel CT3 SPA
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
```

```
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 applique, 2 interface resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions alarm present
  DSU mode 0, bandwidth 44210 Kbit, scramble 0, VC 0
```

The following example shows the output for a serial interface for the first T1 on a channelized T3:

```
Router# show interface serial3/0/1/1:0
Serial3/0/1/1:0 is administratively down, line protocol is down
  Hardware is Channelized/ClearChannel CT3 SPA
  MTU 1500 bytes, BW 832 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions alarm present
  VC 1: timeslot(s): 2-14, Transmitter delay 0, non-inverted data
```

Specifying the Interface Address on a SPA

SPA interface ports begin numbering with "0" from left to right. Single-port SPAs use only the port number 0. To configure or monitor SPA interfaces, you need to specify the physical location of the SIP, SPA, and interface in the CLI. The interface address format is *slot*/*subslot*/*port*, where:

- *slot*—Specifies the chassis slot number in the Cisco 7600 series router where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA.

The following example shows how to specify the first interface (0) on a SPA installed in the first subslot of a SIP (0) installed in chassis slot 3:

```
Router(config)# interface serial 3/0/0
```

This command shows a serial SPA as a representative example, however the same *slot*/*subslot*/*port* format is similarly used for other SPAs (such as ATM and POS) and other non-channelized SPAs.

For the 4-Port Channelized T3 SPA, the interface address format is *slot/subslot/port/t1-number*:*channel-group*, where:

- *t1-number*—Specifies the logical T1 number in channelized mode.
- *channel-group*—Specifies the logical channel group assigned to the timeslots within the T1 link.

For more information about identifying slots and subslots, see the "Identifying Slots and Subslots for SIPs, SSCs, and SPAs" section on page 4-2.

Optional Configurations

There are several standard, but optional configurations that might be necessary to complete the configuration of your serial SPA.

- Configuring the Data Service Unit Mode, page 19-9
- Configuring Maintenance Data Link, page 19-10
- Configuring Encapsulation, page 19-13
- Configuring T3 Framing, page 19-14
- Configuring FDL, page 19-15
- Configuring Scramble, page 19-16
- Configuring Multilink Point-to-Point Protocol (Hardware-based), page 19-17
- Configuring MLFR for T1/E1, page 19-20
- Configuring Multipoint Bridging, page 19-22
- Configuring Bridging Control Protocol Support, page 19-22
- Configuring BCP on MLPPP, page 19-22
- FRF.12 Guidelines, page 19-24
- LFI Guidelines, page 19-24
- HW MLPPP LFI Guidelines, page 19-24
- FRF.12 LFI Guidelines, page 19-24
- Configuring QoS Features on Serial SPAs, page 19-25

Configuring the Data Service Unit Mode

Configure the SPA to connect with customer premise Data Service Units (DSUs) by setting the DSU mode. Subrating a T3 or E3 interface reduces the peak access rate by limiting the data transfer rate. To configure the Data Service Unit (DSU) mode, use the following commands.

z

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **interface serial** *slot/subslot/port*	Selects the controller to configure and enters controller configuration mode. • *slot/subslot/port*—Specifies the location of the controller. See: Specifying the Interface Address on a SPA, page 19-7
Step 3	Router(config-if)# **dsu mode** {**0** | **1** | **2** | **3** | **4**}	Specifies the interoperability mode used by the T3 controller. • **0**—Connects a T3 controller to another T3 controller or to a Digital Link DSU. Bandwidth range is from 300 to 44210 kbps. This is the default. • **1**—Connects a T3 controller to a Kentrox DSU. Bandwidth range is from 1500 to 35000, or 44210 kbps. **Note** If the bandwidth is set between 35000–44210 kbps, an error message is displayed. • **2**—Connects a T3 controller to a Larscom DSU. Bandwidth range is from 3100 to 44210 kbps. • **3**—Connects a T3 controller to an Adtran T3SU 300. Bandwidth range is from 75 to 44210 kbps. • **4**—Connects a T3 controller to a Verilink HDM 2182. Bandwidth range is from 1500 to 44210 kbps.
Step 4	Router(config-if)# **dsu bandwidth** *kbps*	Specifies the maximum allowable bandwidth. • *kbps*—Bandwidth range is from 1 to 44210 kbps.

Verifying DSU Mode

Use the **show controllers serial** command to display the DSU mode of the controller:

```
Router# show controllers serial
Serial3/1/0 -
   Framing is c-bit, Clock Source is Internal
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 0, since reset 0
   Data in current interval (0 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     0 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
     0 Severely Errored Line Secs
     0 Far-End Errored Secs, 0 Far-End Severely Errored Secs
     0 CP-bit Far-end Unavailable Secs
     0 Near-end path failures, 0 Far-end path failures
     0 Far-end code violations, 0 FERF Defect Secs
     0 AIS Defect Secs, 0 LOS Defect Secs
```

```
   Transmitter is sending AIS.
.
.
.
```

Configuring Maintenance Data Link

MDL messages are used to communicate identification information between local and remote ports. The type of information included in MDL messages includes the equipment identification code (EIC), location identification code (LIC), frame identification code (FIC), unit, Path Facility Identification (PFI), port number, and Generator Identification numbers. To configure Maintenance Data Link (MDL), use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.

Command	Purpose
Router(config)# **controller t3** *slot/subslot/port*	Selects the controller to configure and enters controller configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See: Specifying the Interface Address on a SPA, page 19-7
Router(config-controller)# **mdl** [**string** {**eic** | **fic** | **generator** | **lic** | **pfi** | **port** | **unit**} *string*}] | [**transmit** {**idle-signal** | **path** | **test-signal**}]	Configures the MDL message. • **string eic**—Specifies the Equipment Identification Code; can be up to 10 characters. • **string fic**—Specifies the Frame Identification Code; can be up to 10 characters. • **string generator**—Specifies the Generator number string sent in the MDL Test Signal message; can be up to 38 characters. • **string lic**— Specifies the Location Identification Code; can be up to 11 characters. • **string pfi**—Specifies the Path Facility Identification Code sent in the MDL Path message; can be up to 38 characters. • **string port**—Specifies the Port number string sent in the MDL Idle Signal message; can be up to 38 characters. • **string unit**—Specifies the Unit Identification Code; can be up to 6 characters. • **transmit idle-signal**—Enable MDL Idle-Signal message transmission • **transmit path**—Enable MDL Path message transmission. • **transmit test-signal**—Enable MDL Test-Signal message transmission.

Verifying MDL

Use the **show controller** command to display the MDL settings:

```
Router# show controller t3 3/0/0
T3 3/0/0 is down.  Hardware is 2 ports CT3 SPA
  ATLAS FPGA version: 0, FREEDM336 version: 0
  TEMUX84(1) version: 0, TEMUX84(1) version: 0
  SUBRATE FPGA version: 0
  Applique type is Subrate T3
  Receiver has loss of signal.
  MDL transmission is enabled
     EIC: new, LIC: US, FIC: 23, UNIT: myunit
     Path FI: test pfi
     Idle Signal PORT_NO: New-port
     Test Signal GEN_NO: test-message
  FEAC code received: No code is being received
  Framing is C-BIT Parity, Line Code is B3ZS, Clock Source is Line
  Equipment customer loopback
  Data in current interval (869 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation, 0 P-bit Err Secs
     0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
     869 Unavailable Secs, 0 Line Errored Secs
     0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
     0 Severely Errored Line Secs
     0 Far-End Errored Secs, 0 Far-End Severely Errored Secs
     869 CP-bit Far-end Unavailable Secs
     0 Near-end path failures, 0 Far-end path failures
     0 Far-end code violations, 0 FERF Defect Secs
     0 AIS Defect Secs, 870 LOS Defect Secs
```

Configuring Encapsulation

When traffic crosses a WAN link, the connection needs a Layer 2 protocol to encapsulate traffic. To set the encapsulation method, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Channelized Router(config)# **interface serial** *slot/subslot/port/t1-number:channel-group* **Un-channelized** Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure and enters interface configuration mode. • Channelized: *slot/subslot/port/t1-number:channel-group*—Specifies the location of the interface. See: Specifying the Interface Address on a SPA, page 19-7 • Un-channelized: *slot/subslot/port*—Specifies the location of the interface. See: Specifying the Interface Address on a SPA, page 19-7
Router(config-if)# **encapsulation** {**hdlc** | **ppp** | **frame-relay**}	Set the encapsulation method on the interface. • **hdlc**—High-Level Data Link Control (HDLC) protocol for serial interface. This is the default. • **ppp**—Point-to-Point Protocol (PPP) (for serial interface). • **frame-relay**—Frame Relay (for serial interface).

Verifying Encapsulation

Use the **show interface serial** command to display the encapsulation method:

```
Router# show interface serial3/0/0
Serial3/0/0 is down, line protocol is down
  Hardware is Channelized/ClearChannel CT3 SPA
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 applique, 2 interface resets
```

```
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions alarm present
  DSU mode 0, bandwidth 44210 Kbit, scramble 0, VC 0
```

Configuring T3 Framing

To set the T3 framing type, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure and enters interface configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See: "Specifying the Interface Address on a SPA" section on page 19-7
Router(config-if)# **framing** {**c-bit** | **m13**}	Specifies the framing type in unchannelized mode. • **c-bit**—Specifies C-bit parity framing. This is the default. • **m13**—Specifies DS3 Framing M13 (same as M23).

Verifying Framing

Use the **show controller** command to display the framing type:

```
Router# show controller t3 3/0/0
T3 3/0/0 is down.  Hardware is 2 ports CT3 SPA
  ATLAS FPGA version: 0, FREEDM336 version: 0
  TEMUX84(1) version: 0, TEMUX84(1) version: 0
  SUBRATE FPGA version: 0
  Applique type is Subrate T3
  Receiver has loss of signal.
  Framing is M13, Line Code is B3ZS, Clock Source is Line
  Equipment customer loopback
  Data in current interval (656 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation, 0 P-bit Err Secs
     0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
     666 Unavailable Secs, 0 Line Errored Secs
     0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
     0 Severely Errored Line Secs
     0 Far-End Errored Secs, 0 Far-End Severely Errored Secs
     0 CP-bit Far-end Unavailable Secs
     0 Near-end path failures, 0 Far-end path failures
     0 Far-end code violations, 0 FERF Defect Secs
     0 AIS Defect Secs, 666 LOS Defect Secs
```

Configuring FDL

Facility Data Link (FDL) is a far-end performance reporting tool. In ansi mode, you can enable 1-second transmissions of performance reports on both ends of the T1 connection. To configure FDL, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **controller t3** *slot/subslot/port*	Selects the controller to configure and enters controller configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See: "Specifying the Interface Address on a SPA" section on page 19-7
Router(config-controller)# **t1** *number* **fdl** {**ansi**}	(Optional) Enables FDL. • *number*—Specifies the T1 channel number. • **ansi**—Specifies the FDL bit per the ANSI T1.403 specification.

Verifying FDL

Use the **show controller** command to display the FDL setting:

```
Router# show controller t3 3/0/1/1
T3 3/0/1 is down.  Hardware is 2 ports CT3 SPA
  ATLAS FPGA version: 0, FREEDM336 version: 0
  TEMUX84(1) version: 0, TEMUX84(1) version: 0
  SUBRATE FPGA version: 0
  Applique type is Channelized T3
  Receiver has loss of signal.
  Framing is M23, Line Code is B3ZS, Clock Source is Internal
  Equipment customer loopback
  Data in current interval (456 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation, 0 P-bit Err Secs
     0 P-bit Severely Err Secs, 0 Severely Err Framing Secs
     456 Unavailable Secs, 0 Line Errored Secs
     0 C-bit Errored Secs, 0 C-bit Severely Errored Secs
     0 Severely Errored Line Secs
     0 Far-End Errored Secs, 0 Far-End Severely Errored Secs
     0 CP-bit Far-end Unavailable Secs
     0 Near-end path failures, 0 Far-end path failures
     0 Far-end code violations, 0 FERF Defect Secs
     0 AIS Defect Secs, 456 LOS Defect Secs

  T1 1 is down
  timeslots: 2-14
  FDL per ANSI T1.403 and AT&T 54016 spec.
  Configured for FDL remotely line looped (bell)
  Transmitter is sending LOF Indication.
  Receiver is getting AIS.
  Framing is ESF, Clock Source is Line
  BERT running on timeslots 2,3,4,5,6,7,8,9,10,11,12,13,14,
  BERT test result (running)
     Test Pattern : All 1's, Status : Not Sync, Sync Detected : 0
     Interval : 2 minute(s), Time Remain : 2 minute(s)
     Bit Errors (since BERT started): 0 bits,
```

```
      Bits Received (since BERT started): 0 Kbits
      Bit Errors (since last sync): 0 bits
      Bits Received (since last sync): 0 Kbits
   Data in current interval (703 seconds elapsed):
      0 Line Code Violations, 0 Path Code Violations
      0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
      0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs
      713 Unavail Secs, 0 Stuffed Secs
      357 Near-end path failures, 0 Far-end path failures, 0 SEF/AIS Secs
```

Configuring Scramble

T3 scrambling is used to assist clock recovery on the receiving end. Scrambling is designed to randomize the pattern of 1s and 0s carried in the physical layer frame. Randomizing the digital bits can prevent continuous, nonvariable bit patterns—in other words, long strings of all 1s or all 0s. Several physical layer protocols rely on transitions between 1s and 0s to maintain clocking.

Scrambling can prevent some bit patterns from being mistakenly interpreted as alarms by switches placed between the Data Service Units (DSUs).

To configure scrambling, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port*	Selects the interface to configure and enters interface configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See: "Specifying the Interface Address on a SPA" section on page 19-7
Router(config-if)# **scramble** [**0** | **1**]	Enables scrambling. Scrambling is disabled by default. • Scramble settings: **1**—enabled **0**—disabled

Verifying Scrambling

Use the **show interface serial** command to display the scramble setting:

```
Router# show interface serial3/0/0
Serial3/0/0 is down, line protocol is down
  Hardware is Channelized/ClearChannel CT3 SPA
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
```

```
     0 runts, 0 giants, 0 throttles
              0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 applique, 4 interface resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions alarm present
  DSU mode 0, bandwidth 44210 Kbit, scramble 1, VC 0
```

Configuring Multilink Point-to-Point Protocol (Hardware-based)

Multilink Point to Point Protocol (MLPPP) allows you to combine T1 or E1 lines into a bundle that has the combined bandwidth of multiple T1/E1 lines. You choose the number of bundles and the number of T1 or E1 lines in each bundle.

MLPPP for T1/E1 Configuration Guidelines

The required conditions are:

- Only T1 or E1 links in a bundle
- All links on the same SPA
- Maximum of 12 links in a bundle.

Note Some notes about hardware-based MLPPP:

Only 3 fragmentation sizes are possible 128, 256 and 512 bytes

Fragmentation is enabled by default, default size is 512 bytes

Fragmentation size is configured using the **ppp multilink fragment-delay** command after using the **interface multilink** command. The least of the fragmentation sizes (among the 3 sizes possible) satisfying the delay criteria is configured. (e.g., a 192 byte packet causes a delay of 1 millisecond on a T1 link, so the nearest fragmentation size is 128 bytes.

The **show ppp multilink** command will indicate the mlppp type and the fragmentation size:

```
Router# show ppp multilink
Multilink1, bundle name is Patriot2
Bundle up for 00:00:13
Bundle is Distributed
0 lost fragments, 0 reordered, 0 unassigned
0 discarded, 0 lost received, 206/255 load
0x0 received sequence, 0x0 sent sequence
Member links: 2 active, 0 inactive (max not set, min not set)
Se4/2/0/1:0, since 00:00:13, no frags rcvd
Se4/2/0/2:0, since 00:00:10, no frags rcvd
Distributed fragmentation on. Fragment size 512.  Multilink in Hardware.
```

Fragmentation is disabled explicitly by using the **no ppp multilink fragmentation** command after using the **interface multilink** command.

Create a Multilink Bundle

To create a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** *group-number*	Creates a multilink interface and enter multilink interface mode. • *group-number*—The group number for the multilink bundle.
Router(config-if)# **ip address** *address mask*	Sets the IP address for the multilink group. • *address*—The IP address. • *mask*—The IP netmask.

Assign an interface to a Multilink Bundle

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port/t1-number***:***channel-group*	Selects the interface to configure and enters interface configuration mode. See: "Specifying the Interface Address on a SPA" section on page 19-7 • *slot/subslot/port/t1-number***:***channel-group*—Selec t the interface to configure.
Router(config-if)# **encapsulation ppp**	Enables PPP encapsulation.
Router(config-if)# **multilink-group** *group-number*	Assigns the interface to a multilink bundle. • *group-numbe*r—The multilink group number for the T1 or E1 bundle.
Router(config-if)# **ppp multilink**	Enables multilink PPP on the interface.
Repeat these commands for each interface you want to assign to the multilink bundle.	

Configuring fragmentation size on an MLPPP Bundle (optional)

To configure the fragmentation size on a multilink ppp bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** *slot/subslot/port/t1-number***:***channel-group*	Creates a multilink interface and enters multilink interface mode. • group-number—The group number for the multilink bundle. Range 1-2147483647
Router(config-if)# **ppp multilink fragment-delay** *delay*	Sets the fragmentation size satisfying the configured delay on the multilink bundle. • delay—delay in milliseconds

Disabling the fragmentation on an MLPPP Bundle (optional)

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** *group-number*	Creates a multilink interface and enters multilink interface mode. • group-number—The group number for the multilink bundle. Range 1-2147483647
Router(config-if)# **no ppp multilink fragmentation**	Disables the fragmentation on the multilink bundle.

Verifying Multilink PPP

Use the **show ppp multilink** command to verify the PPP multilinks:

```
router# show ppp multilink
Multilink1, bundle name is mybundle
    Bundle up for 01:40:50
    Bundle is Distributed
    0 lost fragments, 0 reordered, 0 unassigned
    0 discarded, 0 lost received, 1/255 load
    0x0 received sequence, 0x0 sent sequence
Member links: 5 active, 0 inactive (max not set, min not set)
    Se6/0/0/1:0, since 01:40:50, no frags rcvd
    Se6/0/1/1:0, since 01:40:09, no frags rcvd
    Se6/0/3/1:0, since 01:15:44, no frags rcvd
    Se6/0/4/1:0, since 01:03:17, no frags rcvd
    Se6/0/6/1:0, since 01:01:06, no frags rcvd
    Se6/0/6:0, since 01:01:06, no frags rcvd
```

Configuring MLFR for T1/E1

Multilink Frame Relay (MLFR) allows you to combine T1/E1 lines into a bundle that has the combined bandwidth of multiple T1/E1 lines. You choose the number of bundles and the number of T1/E1 lines in each bundle. This allows you to increase the bandwidth of your network links beyond that of a single T1/E1 line.

MLFR for T1/E1 Configuration Guidelines

MLFR will function in hardware if all of the following conditions are met:

- Only T1 or E1 member links
- All links are on the same SPA
- Maximum of 12 links in a bundle

Create a Multilink Bundle

To create a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface mfr** *number*	Configures a multilink Frame Relay bundle interface. • *number*—The number for the Frame Relay bundle.
Router(config-if)# **frame-relay multilink bid** *name*	(Optional) Assigns a bundle identification name to a multilink Frame Relay bundle. • *name*—The name for the Frame Relay bundle. **Note** The bundle identification (BID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shut** and **no shut** commands in interface configuration mode.

Assign an Interface to a Multilink Bundle

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** *slot/subslot/port***:***channel-group*	Selects the interface to assign. • *slot/subslot/port:channel-group*—Specifies the location of the interface."Specifying the Interface Address on a SPA" section on page 19-7
Router(config-if)# **encapsulation frame-relay mfr** *number* [*name*]	Creates a multilink Frame Relay bundle link and associates the link with a bundle. • *number*—The number for the Frame Relay bundle. • *name*—The name for the Frame Relay bundle.
Router(config-if)# **frame-relay multilink lid** *name*	(Optional) Assigns a bundle link identification name with a multilink Frame Relay bundle link. • *name*—The name for the Frame Relay bundle. **Note** The bundle link identification (LID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shut** and **no shut** commands in interface configuration mode.
Router(config-if)# **frame-relay multilink hello** *seconds*	(Optional) Configures the interval at which a bundle link will send out hello messages. The default value is 10 seconds. • *seconds*—Number of seconds between hello messages sent out over the multilink bundle.
Router(config-if)# **frame-relay multilink ack** *seconds*	(Optional) Configures the number of seconds that a bundle link will wait for a hello message acknowledgment before resending the hello message. The default value is 4 seconds. • *seconds*—Number of seconds a bundle link will wait for a hello message acknowledgment before resending the hello message.
Router(config-if)# **frame-relay multilink retry** *number*	(Optional) Configures the maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment. The default value is 2 tries. • *number*—Maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment.

Verifying Multilink Frame Relay

Use the show frame-relay multilink detailed command to verify the Frame Relay multilinks:

```
Router# show frame-relay multilink detailed
Bundle: MFR49, State = down, class = A, fragmentation disabled
 BID = MFR49
 No. of bundle links = 1, Peer's bundle-id =
 Bundle links:

  Serial6/0/0:0, HW state = up, link state = Add_sent, LID = test
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
    Peer LID = , RTT = 0 ms
    Statistics:
    Add_link sent = 21, Add_link rcv'd = 0,
    Add_link ack sent = 0, Add_link ack rcv'd = 0,
    Add_link rej sent = 0, Add_link rej rcv'd = 0,
    Remove_link sent = 0, Remove_link rcv'd = 0,
    Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
    Hello sent = 0, Hello rcv'd = 0,
    Hello_ack sent = 0, Hello_ack rcv'd = 0,
    outgoing pak dropped = 0, incoming pak dropped = 0
```

Configuring Multipoint Bridging

Multipoint bridging (MPB) enables the connection of multiple ATM PVCs, Frame Relay PVCs, BCP ports, and WAN Gigabit Ethernet subinterfaces into a single broadcast domain (virtual LAN), together with the LAN ports on that VLAN. This enables service providers to add support for Ethernet-based Layer 2 services to the proven technology of their existing ATM and Frame Relay legacy networks. Customers can then use their current VLAN-based networks over the ATM or Frame Relay cloud. This also allows service providers to gradually update their core networks to the latest Gigabit Ethernet optical technologies, while still supporting their existing customer base.

For MPB configuration guidelines and restrictions and feature compatibility tables, see the "Configuring Multipoint Bridging" section on page 4-23 of Chapter 4, "Configuring the SIPs and SSC."

Configuring Bridging Control Protocol Support

The Bridging Control Protocol (BCP) enables forwarding of Ethernet frames over SONET networks and provides a high-speed extension of enterprise LAN backbone traffic through a metropolitan area. The implementation of BCP on the SPAs includes support for IEEE 802.1D, IEEE 802.1Q Virtual LAN (VLAN), and high-speed switched LANs.

For BCP configuration guidelines and restrictions and feature compatibility tables, see the "Configuring PPP Bridging Control Protocol Support" section on page 4-35 of Chapter 4, "Configuring the SIPs and SSC."

Configuring BCP on MLPPP

BCP on MLPPP Configuration Guidelines

- Only Distributed MLPPP is supported
- Only channelized interfaces allowed, and member links must be from the same controller card
- Only trunk port BCP is supported on MLPPP
- Bridging can be configured only on the bundle interface

Note BCP on MLPPP operates only in trunk mode.

Configuring BCP on MLPPP Trunk Mode

To configure BCP on MLPPP trunk mode, perform these steps:

	Command	Purpose
Step 1	Router(config)# **interface multilink**	Selects the multilink interface.
Step 2	Router(config-if)# **switchport**	Puts an interface that is in Layer 3 mode into Layer 2 mode for Layer 2 configuration.
Step 3	Router(config-if)#**switchport trunk allowed vlan 100**	By default, no VLANs are allowed. Use this command to explicitly allow VLANs; valid values for *vlan-list* are from 1 to 4094.
Step 4	Router(config-if)#**switchport mode trunk**	Configures the router port connected to the switch as a VLAN trunk port.
Step 5	Router(config-if)#**switchport nonegotiate**	Puts the LAN port into permanent trunking mode but prevents the port from generating DTP frames
Step 6	Router(config-if)#**no ip address**	
Step 7	Router(config-if)#**ppp multilink**	Enables this interface to support MLP.
Step 8	Router(config-if)#**multilink-group 1**	Assigns this interface to the multilink group.
Step 9	Router(config-if)# **interface Serial1/0/0.1/1/1/1:0**	Designates a serial interface as a multilink bundle.
Step 10	Router(config-if)# **no ip address**	Unassigns the IP address.
Step 11	Router(config-if)#**encapsulation ppp**	Enables PPP encapsulation.
Step 12	Router(config-if)#**ppp multilink**	Enables this interface to support MLP.
Step 13	Router(config-if)# **multilink-group 1**	Assigns this interface to the multilink group 1.
Step 14	Router(config-if)#**interface Serial1/0/0.1/1/1/2:0**	Designates a serial interface as a multilink bundle.
Step 15	Router(config-if)#**no ip address**	Unassigns the IP address.
Step 16	Router(config-if)#**encapsulation ppp**	Enables PPP encapsulation.
Step 17	Router(config-if)#**ppp multilink**	Enables this interface to support MLP.
Step 18	Router(config-if)# **multilink-group 1**	Assigns this interface to the multilink group 2.
Step 19	Router(config-if)# **shutdown**	Shuts down an interface.
Step 20	Router(config-if)# **no shutdown**	Reopens an interface.
Step 21	Router(config-if)# **switchport trunk allowed vlan** vlan-list	By default, no VLANs are allowed. Use this command to explicitly allow VLANs; valid values for *vlan-list* are from 1 to 4094.

Verifying BCP on MLPPP Trunk Mode

To display information about Multilink PPP, use the show ppp multilink command in EXEC mode.

Command	Purpose
Router(config-if)# **show ppp multilink**	Displays information on a multilink group.

The following shows an example of **show ppp multilink**:

```
Router# show ppp multilink

Multilink1, bundle name is group 1
Bundle is Distributed
0 lost fragments, 0 reordered, 0 unassigned, sequence 0x0/0x0 rcvd/sent
0 discarded, 0 lost received, 1/255 load
Member links: 4 active, 0 inactive (max no set, min not set)
Serial1/0/0/:1
Serial1/0/0/:2
Serial1/0/0/:3
Serial1/0/0/:4
```

FRF.12 Guidelines

FRF.12 functions in hardware. Note the following:

- The fragmentation is configured at the main interface
- Only 3 fragmentation sizes are available - 128 bytes, 256 bytes, and 512 bytes.

LFI Guidelines

LFI can function two ways - using FRF.12 or MLPPP. MLPPP LFI can be done in both hardware and software while FRF.12 LFI is done only in hardware.

HW MLPPP LFI Guidelines

LFI using MLPPP will function only in hardware if there is just one member link in the MLPPP bundle. The link can be a fractional T1 or full T1. Note the following:

- The **ppp multilink interleave** command needs to be configured to enable interleaving.
- Only three fragmentation sizes are supported - 128 bytes, 256 bytes, and 512 bytes.
- Fragmentation is enabled by default, the default size being 512 bytes.
- A policy-map having a priority class needs to applied to main interface.
- When hardware-based LFI is enabled, fragmentation counters are not displayed.

FRF.12 LFI Guidelines

LFI using FRF.12 is always done is hardware. Note the following:

- The fragmentation is configured at the main interface
- Only 3 fragmentation sizes are available - 128 bytes, 256 bytes, and 512 bytes.
- A policy-map having a priority class needs to applied to main interface.

Configuring QoS Features on Serial SPAs

The SIPs and SPAs support many QoS features using modular QoS CLI (MQC) configuration. For information about the QoS features supported by the serial SPAs, see the "Configuring QoS Features on a SIP" section on page 4-61 of Chapter 4, "Configuring the SIPs and SSC."

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), use the following command in privileged EXEC configuration mode:

Command	Purpose
Router# **copy running-config startup-config**	Writes the new configuration to NVRAM.

For more information about managing configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

Verifying the Interface Configuration

Besides using the **show running-configuration** command to display your Cisco 7600 series router configuration settings, you can use the **show interfaces serial** and the **show controllers serial** commands to get detailed information on a per-port basis for your 2-Port and 4-Port Clear Channel T3/E3 SPA.

Verifying Per-Port Interface Status

To find detailed interface information on a per-port basis for the 2-Port and 4-Port Channelized T3 SPA, use the **show interfaces serial** command.

The following example provides sample output for the serial interface on an un-channelized T3:

```
Router# show interface serial3/0/0
Serial3/0/0 is down, line protocol is down
  Hardware is Channelized/ClearChannel CT3 SPA
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
```

```
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 applique, 4 interface resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions alarm present
  DSU mode 0, bandwidth 44210 Kbit, scramble 1, VC 0
```

The following example provides sample output for the serial interface on a channelized T3:

```
Router# show interface serial3/0/1/1:0
Serial3/0/1/1:0 is down, line protocol is down
  Hardware is Channelized/ClearChannel CT3 SPA
  MTU 1500 bytes, BW 832 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions alarm present
  VC 1: timeslot(s): 2-14, Transmitter delay 0, non-inverted data
```

To find detailed status and statistical information on a per-port basis for the 2-Port and 4-Port Clear Channel T3/E3 SPA, use the **show controllers serial** command.

The following example provides sample controller statistics for the third port on the SPA located in the first subslot of the SIP-200 that is installed in slot 5 of a Cisco 7609 router:

```
show controller serial 5/0/2
Serial5/0/2 -
   Framing is c-bit, Clock Source is Line
   Bandwidth limit is 44210, DSU mode 0, Cable length is 10
   rx FEBE since last clear counter 0, since reset 0
   Data in current interval (807 seconds elapsed):
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 306 Unavailable Secs
     500 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 1:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     564 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 2:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     564 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 3:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
```

```
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     562 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
  Data in Interval 4:
     0 Line Code Violations, 0 P-bit Coding Violation
     0 C-bit Coding Violation
     0 P-bit Err Secs, 0 P-bit Sev Err Secs
     0 Sev Err Framing Secs, 0 Unavailable Secs
     560 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs
    .
    .
    .
    Total Data (last 44 15 minute intervals):
         0 Line Code Violations, 0 P-bit Coding Violation,
         0 C-bit Coding Violation,
         0 P-bit Err Secs, 0 P-bit Sev Err Secs,
         0 Sev Err Framing Secs, 0 Unavailable Secs,
         24750 Line Errored Secs, 0 C-bit Errored Secs, 0 C-bit Sev Err Secs

      Transmitter is sending AIS.

      Receiver has loss of signal.

       40434 Sev Err Line Secs, 0 Far-End Err Secs, 0 Far-End Sev Err Secs
       0 P-bit Unavailable Secs, 0 CP-bit Unavailable Secs
       0 CP-bit Far-end Unavailable Secs
       0 Near-end path failures, 0 Far-end path failures

      No FEAC code is being received
     MDL transmission is disabled
```

Configuration Examples

This section includes the following configuration examples:

- DSU Configuration Example, page 19-27
- MDL Configuration Example, page 19-28
- Encapsulation Configuration Example, page 19-28
- Framing—Unchannelized Mode Configuration Example, page 19-28
- Facility Data Link Configuration Example, page 19-29
- Scrambling Configuration Example, page 19-29
- Creating a Multilink Bundle Configuration Example, page 19-29
- Assigning a T1 Interface to a Multilink Bundle Configuration Example, page 19-29

DSU Configuration Example

The following example sets the DSU mode on interface port 0 on slot 4, subslot 1.

```
! Specify the interface and enter interface configuration mode.
!
Router(config-int)# interface t3 4/1/0
!
!Specifies the interoperability mode used by the T3 interface.
!
```

```
Router(config-int)# dsu mode 2
!
!Specifies the maximum allowable bandwidth.

Router(config-int)# dsu bandwidth 23000
```

MDL Configuration Example

The following example configures the MDL strings on controller port 0 on slot 4, subslot 1.

```
! Enter controller configuration mode.
!
Router(config)# controller t3 4/1/0
!
! Specify the mdl strings.
!
Router(config-controller)# mdl string eic beic
Router(config-controller)# mdl string lic beic
Router(config-controller)# mdl string fic bfix
Router(config-controller)# mdl string unit bunit
Router(config-controller)# mdl string pfi bpfi
Router(config-controller)# mdl string port bport
Router(config-controller)# mdl string generator bgen
Router(config-controller)# mdl transmit path
Router(config-controller)# mdl transmit idle-signal
Router(config-controller)# mdl transmit test-signal
```

Encapsulation Configuration Example

The following example configures encapsulation on a channelized T1 interface.

```
! Specify the interface to configure and enter interface configuration mode.
!
Router(config)# interface serial 4/1/1/1:0
!
! Specify the encapsulation method.
!
Router(config-if)# encapsulation ppp
```

The following example configures encapsulation and framing on a un-channelized T3 interface.

```
! Specify the interface to configure and enter interface configuration mode.
!
Router(config)# interface serial 4/1/1
!
! Specify the encapsulation method.
!
Router(config-if)# encapsulation ppp
```

Framing—Unchannelized Mode Configuration Example

The following example configures framing on an un-channelized T3 interface.

```
! Specify the interface to configure and enter interface configuration mode.
!
Router(config)# interface serial 4/1/1
!
! Specify the framing type.
!
```

```
Router(config-if)# framing m13
```

Facility Data Link Configuration Example

The following example configures FDL on a channelized T1 interface.

```
! Specify the controller to configure and enter controller configuration mode.
!
Router(config)# controller t3 3/1/0
!
! Specify the T1 controller and set the FDL bit.
!
Router(config-controller)# t1 1 fdl ansi
```

Scrambling Configuration Example

The following example configures scrambling on the T3 interface:

```
! Enter global configuration mode.
!
Router# configure terminal
!
! Specify the interface to configure and enter interface configuration mode.
!
Router(config)# interface serial 4/1/3
!
! Enable scrambling.
!
Router(config-if)# scrambling
```

Creating a Multilink Bundle Configuration Example

```
! ! Enter global configuration mode.
!
Router# configure terminal
!
! Create a multilink interface and enter interface configuration mode.
!
Router(config)# interface multilink 1
!
! Specify the IP address for the interface.
!
Router(config-if)# ip address 123.345.678.21 255.255.255.0
!
```

Assigning a T1 Interface to a Multilink Bundle Configuration Example

```
! ! Enter global configuration mode.
!
Router# configure terminal
!
! Specify the T1 interface and enter interface configuration mode.
!
Router(config)# interface serial 1/0/1/1:0
!
! Specify PPP encapsulation.
```

```
!
Router(config-if)# encapsulation ppp
!
! Specify the multilink bundle the T1 will belong to.
!
Router(config-if)# multilink-group 1
!
```

CHAPTER 20

Configuring the 1-Port Channelized OC-3/STM-1 SPA

This chapter provides information about configuring the 1-Port Channelized OC-3/STM-1 SPA on Cisco 7600 series routers. It includes the following sections:

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Configuration Tasks

This section describes how to configure the 1-Port Channelized OC-3/STM-1 SPA for the Cisco 7600 series routers and includes information about verifying the configuration. This document shows how to configure the 1-Port Channelized OC-3/STM-1 SPA in either SONET or SDH framing modes.

It includes the following topics:

Required Configuration Tasks

This section lists the required configuration steps to configure the 1-Port Channelized OC-3/STM-1 SPA. Some of the required configuration commands implement default values that might be appropriate for your network. If the default value is correct for your network, then you do not need to configure the command.

- Selection of Physical Port and Controller configuration

- Interface Naming
- SONET mode Configuration
- SDH mode Configuration
- Verifying Interface Configuration

Note To better understand the address format used to specify the physical location of the Cisco 7600 SIP-200, SPA, and interfaces, see the: "Selection of Physical Port and Controller configuration" section on page 20-2.

Selection of Physical Port and Controller configuration

To select the physical port and controller configuration, use the following command:

controller sonet *slot* / *subslot* / *port*

If the 1-Port Channelized OC-3/STM-1 SPA sits in subslot 0 of a Cisco 7600 SIP-200 in slot3, the 1-Port Channelized OC-3/STM-1 SPA port would be identified as controller SONET 3/0/0. Since there is only 1 port on a 1-Port Channelized OC-3/STM-1 SPA, the port number is always 0.

Interface Naming

Interface names are automatically generated, and the format will be dependent on the mode each particular linecard is operating on. The name format of the serial interface created are listed below.

SONET mode

- If framing is SONET and mode is vt-15:

  interface serial [*slot* / *subslot* / *port*].[*sts*1/ *ds1* / *t1*]:[*channel-group*]

- If framing is SONET and mode is CT3

  interface serial [*slot* / *subslot* / *port*].[*sts*1 / *ds1* / *ds1*]:[*channel-group*]

- If framing is SONET and mode is CT3-E1:

  interface serial [*slot* / *subslot* / *port*].[*sts*1 / *ds1* / *e1*]:[*channel-group*]

- If framing is SONET and mode is T3:

  interface serial [*slot* / *subslot* / *port.sts1*]

SDH mode

If the aug mapping is au-4, the au-4 value is always 1; if the aug mapping is au-3, then the only supported mode is c-11 (carrying a T1).

- If SDH-AUG mapping is au-4 and if the tug-3 is mode t3/e3:

  interface serial [*slot / subslot / 0.1 / <tug-3>*]

- If SDH-AUG mapping is au-3:

  interface serial [*slot* / *subslot / port* / *au-3 / <tug-2> / t1*]:[*channel-group*]

- If framing is SDH with ct-12 mode:

  interface serial [*slot/ subslot / 0.1 / <tug-3> / <tug-2> / e1*]:[*channel-group*]

- If framing is SDH with c-11 mode:

  interface serial [*slot / subslot / 0.<au-3> / <tug-2> / t1*]*:* [*channel-group*]

For channelized T3 mode

- If framing is SONET or SDH with au-3:

  interface serial [*slot* | *subslot* | *port*] [*ds3*| *DS1*]*:*[*channel-group*]

Selection of Physical Port and controller Configuration—SONET mode

To create the interface for the 1-Port Channelized OC-3/STM-1 SPA, complete these steps:

	Command	Purpose
Step 1	Router(config)# **controller sonet** *slot/subslot/port*	Select the controller to configure and enter controller configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See the: "Selection of Physical Port and Controller configuration" section on page 20-2 **Note** The port number is always zero on the 1-Port Channelized OC-3/STM-1 SPA.

SONET mode Configuration

To configure the SONET controller, complete these steps:

Step 1	For SONET controllers: Router(config-controller)# **framing** {**sonet** | **sdh**}	Selects the framing type. • **sonet**—Specifies SONET as the frame type. This is the default. • **sdh**—Specifies sdh as the frame type.
Step 2	Router(config-controller)# **clock source** {**internal** | **line**}	Sets the clock source. **Note** The clock source is set to internal if the opposite end of the connection is set to line and the clock source is set to line if the opposite end of the connection is set to internal. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used. This is the default for T1 and E1.

Step 3	Router(config-controller)# **[no] loopback** {**local** | **network**]	Enables or disables loopback mode on a sonet controller. • **local loopback**—loops data from the transmit path to the receive path. • **network loopback**—loops data received on the external port to the transmit path and back out the external port. Default is disabled loopback.
Step 4	In SONET framing: Router(config-controller)#**sts-1 sts1-#**	**sts-1 #**—Specifies the SONET STS level.
Step 5	**[no] mode** {**vt-15** | **ct3** | **t3** | **ct3-e1**}	Specifies the mode of operation of a STS-1 path: • **vt-15**—A STS-1 is divided into 7 vtg. Each vtg then divided into 4 VT1.5's, each carrying a T1. • **ct3**—A STS-1 carry a DS3 signal divided into 28 T1s (PDH) • **t3—**STS-1 or AU-4/TUG3 carries a unchannelized (clear channel) T3 • **ct3**—The channelized T3 is carrying E1 circuits
Step 6	• sts1 carries T1s (sonet - vt): router(config-controller-sts1)# **mode vt-15** or • sts1 carries T1s (sonet - ds3 down to ds1): router(config-controller-sts1)# **mode ct3** or • sts1 carries DS3(sonet - ds3): router(config-controller-sts1)# **mode t3**	Selects a mode of operation.
Step 7	Router(config-ctrlr-sts1)# **vtg**? <1-7> vtg number <1-7>	• **vtg**—Specifies the vtg number.

Step 8	Router(config-ctrlr-sts1)#**vtg 1 ?** T1 T1 line configuration Router(config-ctrlr-sts1)#v**tg 1** t1 1 chan 0 tim 1 - 3 Router(config-ctrlr-sts1)#**vtg 2** t1 4 chan 0 tim 1 - 2, 5-6 Router(config-ctrlr-sts1)#**vtg 3** t1 # <1-4> t1 line number <1-4>	Configures the T1s on the vtgs. For SONET framing, vtg# range is 1 to 7.
Step 9	Channelized OC-3: vtg <vtg#>... ct3: no prefix There is no channelized E3 mode. The e1# range is from 1 to 3 The t1# range is from 1 to 4. For PDH mode, where a channelized t3 is mapped into the sts-1, the t1# range is from 1 to 28.	Configures channels. Once TUG-3/STS-1 is configured, then one of the parser modes config-ctrlr-{tug3|au3|sts1} can be set.

SDH mode Configuration

To configure SDH mode, complete the following steps:

Step 1	For SDH controllers: Router(config-controller)# **framing** {**sonet** | **sdh**}	Selects the framing type. • **sonet**—Specifies SONET as the frame type. This is the default. • **sdh**—Specifies sdh as the frame type.
Step 2	Router(config-controller)# **aug mapping** {**au-3** | **au-4**}	Configures AUG mapping for SDH only. If the AUG mapping is configured to be au-4, then the following muxing/alignment/mapping will be used: TUG-3 <--> VC-4 <--> AU-4 <--> AUG If the mapping is configured to be au-3, then the following muxing/alignment/mapping will be used: VC-3 <--> AU-3 <--> AUG This command will be available only when sdh framing is configured. Default is **au-4.**
Step 3	If AUG mapping is au-4: au-4 <au-4#> tug-3 <tug-3#> If AUG mapping is au-3: au-3 <au-3#>	Configures TUG-3/AU-3/STS-1. Depending on the framing mode of Sonet or SDH, each STS-1 and each TUG-3/AU-3 of a STM-1 can be configured with this command. Depending on currently configured AUG mapping setting, this command will further specify TUG-3, AU-3 or STS-1 muxing. As result, the CLI command parser will enter into config-ctrlr-tug3, config-ctrlr-au3 or config-ctrlr-sts1 parser mode, which will make only relevant commands visible. The au-4# is 1. The tug-3# range is from 1 to 3. The au-3# range is from 1 to 3. The sts-1# is from 1 to 3.
Step 4	In SDH framing in AU-4 mode: **[no] mode {c-12 | t3 | e3}**	C-11 and c-12 are container level-n (SDH) Channelized T3s. They are types of T3 channels that are subdivided into 28 T1 channels. • **c-12**—Specifies a AU-4/TUG-3 is divided into 7 tug2. Each tug2 then divided into 3 TU12's, each carrying an E1 (C-12). • **c-11**—Specifies a AU-3 is divided into 7 tug2. Each tug2 then divided into 4 TU11's, each carrying a T1 (C-11). • **t3**—Specifies a STS-1 or AU-4/TUG3 carries a unchannelized (clear channel) T3 • **e3**—Specifies a AU-4/TUG3 carries a unchannelized (clear channel) E3

Configure Channelized DS3

To configure channelized DS3 mode, complete the following steps:

Step 1	Router(config)# **controller sonet** *slot/subslot/port*	Select the controller to configure and enter controller configuration mode. • *slot/subslot/port*—Specifies the location of the interface. See the: "Selection of Physical Port and Controller configuration" section on page 20-2
Step 2	Router(config)#sts-1 *sts1-#*	The sts-1# is from 1 to y, y being the Sonet STS level, such as in OC-3.
Step 3	Router(config)# **t3 framing** {**c-bit** | **m23** | **auto-detect**}	Specifies framing mode. • **c-bit**—Specifies C-bit parity framing. • **m23**—Specifies M23 framing. • **auto-detect—**Detects the framing type at the device at the end of the line and switches to that framing type. If both devices are set to auto-detect, c-bit framing is used.
Step 4	Router(config-controller)# **clock source** {**internal** | **line**}	Sets the clock source. **Note** The clock source is set to internal if the opposite end of the connection is set to line and the clock source is set to line if the opposite end of the connection is set to internal. • **internal**—Specifies that the internal clock source is used. • **line**—Specifies that the network clock source is used.
Step 5	Router(config-controller)# **[no] t3 loopback** {**local** | **network [line | payload]** | **remote [line | payload]**}	Enables or disables loopback mode on a SONET controller. • **local loopback**—loops data from the transmit path to the receive path. • **network loopback**—loops data received on the external port to the transmit path and back out the external port. • **Remote loopback**—Applicable only to c-bit framing. Default is no loopback.
Step 6	**[no] t3 mdl string** [**eic** | **fic** | **generator** | **lic** | **pfi** | **port** | **unit**} *string* **[no] t3 mdl transmit** {**path** | **idle-signal** | **test-signal**}	Configures MDL support. • **eic**—Specified equipment ID code • **fic**— frame ID code • **generator**—generator number in MDL test signal • **lic**—location ID code • **pfi—**facility ID code in MDL path message • **port**— port number in MDL idle string message • **unit**—unit code Default is no mdl string and no mdl transmit.

Step 7	**t3 equipment** {**customer** | **network**} **loopback**	Equipment customer loopback enables the port to honor remote loopback request. Equipment network loopback disables this functionality. **Note** Remote loopbacks are only available in c-bit framing mode.
Step 8	**t3 bert pattern** *pattern* **interval** *1-14400*	Enables BERT testing.

DS1 Configuration (Channelized T3 mode)

To configure DS1 complete the following steps:

Step 1	[no] *prefix* **t1** *t1#* **clock source** {**internal | line**}	Configures the clocking source.
Step 2	[no] *prefix* **t1** *t1#* **fdl** *ansi*	Enables the one-second transmission of the remote performance reports via Facility Data Link (FDL) per ANSI T1.403 Note that without this command, FDL will run in ATT mode. ATT mode is not mutually exclusive or different from ANSI mode, ANSI mode is a super-set of ATT mode.
Step 3	[no] *prefix* **t1** *t1#* **framing** {**sf | esf**} [no] *prefix* **t1** *t1#* **yellow** {**detection | generation**}	Enables detection and generation of DS1 yellow alarms
Step 4	[no] *prefix* **t1** *t1#* **shutdown**	
Step 5	[no] prefix **t1** *t1#* **channel-group** *channel-group#* **timeslots** *list-of-timeslots* **speed** [**56 | 64**]	**Note**
Step 6	[no] *prefix* **t1** *t1#* **loopback** {**local | network line | remote {line fdl {ansi | bellcore**} | **payload fdl ansi**}}	**Note** Local network payload loopback is not supported due to TEMUX-84/TEMUX-84E limitations. **Note** Only 6 E1 berts can be performed concurrently due to TEMUX-84/TEMUX-84E limitations.

E1 Configuration (Channelized T3/E3 mode)

E1 configuration must be done in channelized DS3 mode. To configure E1, complete the following steps:

Step 1	[no] *prefix* **e1** *e1# channel-group channel-group#* **timeslots** *list-of-timeslots* **speed** [**56 | 64**]	
Step 2	[no] *prefix* **e1** *e1#* **unframed**	
Step 3	[no] *prefix* **e1** *e1#* **[unframed | framing]** {**crc4 | no-crc4**}	
Step 4	[no] *prefix* **e1** *e1#* **clock source** {**internal | line**}	Configures clock source.
Step 5	[no] *prefix* **e1** *e1#* **national bits** *pattern*	

Step 6	[no] *prefix* **e1** *e1#* **loopback [local | network]**	
Step 7	[no] *prefix* **e1** *e1#* **shutdown**	

BERT Test Configuration

To configure BERT test, complete the following:

Step 1	[no] */* **[e1 | t1] [e1# | t1#] bert pattern** {**2^11 | 2^15 || 2^20 QRSS** } **interval** *time*	Send a BERT pattern on a DS1/E1 line.

Unchannelized E3 Serial Interface Configuration

To configure an unchannelized E3 serial interface, complete the following:

Step 1	[no] **dsu mode** { **cisco | digital-link | kentrox** }	• **cisco**—Specifies cisco as the dsu mode. • **digital-link**—Specifies Digital link as the dsu mode. Range is from 300-34010. • **kentrox**—Specifies kentrox as the dsu mode. Range is 1000-24500, 34010. Default is **cisco**.
Step 2	[no] **dsu bandwidth** *number*	Specifies the maximum allowed bandwidth in Kpbs.
Step 3	[no] **scramble**	Default is no scramble.
Step 4	[no] **national bit** {**0 | 1**}	Default is 0.
Step 5	[no] **crc** {**16 | 32**}	Default is 16 bit (CRC-CITT).
Step 6	[no] **loopback** {**network | local | remote**}	
Step 7	[no] **shutdown**	
Step 8	[no] **bert pattern** *pattern* **interval** *1-14400*	An example of a valid pattern is {2^15 | 2^23 | 0s | 1s }.

Use the **show controllers** command to verify the controller configuration:

```
Router(config)# show controllers t1
T1 6/0/1 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
blarm-trigger is not set
```

```
  Framing is ESF, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (395 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
  Total Data (last 24 hours)
     0 Line Code Violations, 0 Path Code Violations,
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins,
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
```

Verifying Interface Configuration

Use the **show interface serial** command to verify the interface configuration:

```
Router(config)# show interface serial
Serial2/0/0.1/2 unassigned YES TFTP administratively down down
Serial2/1/0.1/1/1:0 unassigned YES unset down down
Serial2/1/0.1/2/4:0 unassigned YES unset down down
Serial2/1/0.1/2/4:1 unassigned YES unset down down
Serial2/1/0.2/1:0 unassigned YES unset down down
Serial2/1/0.2/2:0 unassigned YES unset down down
Serial2/1/0.2/3:0 unassigned YES unset down down
Serial2/1/0.3 unassigned YES unset down down
UUT#sh int Serial2/1/0.1/1/1:0
Serial2/1/0.1/1/1:0 is down, line protocol is down
Hardware is Channelized-T3
MTU 1500 bytes, BW 192 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, crc 16, loopback not set
Keepalive set (10 sec)
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Available Bandwidth 192 kilobits/sec
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 2 interface resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions alarm present
VC 2: timeslot(s): 1-3, Transmitter delay 0, non-inverted data
UUT#sh run | beg 2/1/0
controller SONET 2/1/0
ais-shut
framing sonet
clock source line
overhead j0 1
!
sts-1 1
mode vt-15
vtg 1 t1 1 channel-group 0 timeslots 1-3
vtg 2 t1 4 channel-group 0 timeslots 1-2,5-6
vtg 2 t1 4 channel-group 1 timeslots 3,7,9
!
sts-1 2
mode ct3
t1 1 channel-group 0 timeslots 1-24
t1 2 channel-group 0 timeslots 1-12
t1 3 channel-group 0 timeslots 1
!
```

```
sts-1 3
mode t3
!
controller T3 3/1/0
shutdown
cablelength 224
!
controller T3 3/1/1
shutdown
cablelength 224
!
!
interface Loopback0
ip address 172.10.11.1 255.255.255.255
.
.
```

Optional Configurations

There are several standard, but optional, configurations that might be necessary to complete the configuration of your serial SPA.

Configuring Encapsulation

When traffic crosses a WAN link, the connection needs a Layer 2 protocol to encapsulate traffic. To set the encapsulation method, use the following commands:

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **interface serial** For addressing information, refer to the "Interface Naming" section on page 20-2.	Selects the interface to configure. • *slot/subslot/port:channel-group*—Specifies the location of the interface.
Step 3	Router(config-if)# **encapsulation** *encapsulation-type* **{hdlc | ppp | frame-relay}**	Set the encapsulation method on the interface. • **hdlc**—High-Level Data Link Control (HDLC) protocol for serial interface. This encapsulation method provides the synchronous framing and error detection functions of HDLC without windowing or retransmission. This is the default for synchronous serial interfaces. • **ppp**—PPP (for serial interface). • **frame-relay**—Frame Relay (for serial interface).
Step 4	Router(config-if)# **crc** {**16** | **32**}	Selects the CRC size in bits. • **16**—16-bit CRC. This is the default • **32**—32-bit CRC.

Configuring the CRC Size for T1

The 1-Port Channelized OC-3/STM-1 SPA interface uses a 16-bit cyclic redundancy check (CRC) by default, but also support a 32-bit CRC. CRC is an error-checking technique that uses a calculated numeric value to detect errors in transmitted data. The designators 16 and 32 indicate the length (in bits) of the frame check sequence (FCS). A CRC of 32 bits provides more powerful error detection, but adds overhead. Both the sender and receiver must use the same setting.

CRC-16, the most widely used CRC throughout the United States and Europe, is used extensively with WANs. CRC-32 is specified by IEEE 802 and as an option by some point-to-point transmission standards. It is often used on Switched Multimegabit Data Service (SMDS) networks and LANs.

To set the length of the cyclic redundancy check (CRC) on a T1 interface, use these commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** For addressing information, refer to the "Interface Naming" section on page 20-2.	Selects the interface to configure. • *slot/subslot/port:channel-group*—Specifies the location of the interface.

Configuring FDL

Facility Data Link (FDL) is a 4-kbps channel provided by the Extended Super Frame (ESF) T1 framing format. The FDL performs outside the payload capacity and allows you to check error statistics on terminating equipment without intrusion.

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **controller sonet** *slot/subslot/port* See the "Interface Naming" section on page 20-2.	Selects the controller to configure. • *slot/subslot/port*—Specifies the location of the controller.
Router(config-controller)# **sts-1**	If the framing format was configured for **esf**, configures the format used for Facility Data Link (FDL). • **ansi**—Select ansi for FDL to use the ANSI T1.403 standard.
Router(config-controller)**vtg 1** t1 1 fdl	• **vtg**—Specifies the vtg number

Verifying FDL

Use the **show controllers t1** command to verify the **fdl** setting:

```
router# show controllers t1
T1 6/0/1 is up.
  Applique type is Channelized T1
  Cablelength is long gain36 0db
  No alarms detected.
  alarm-trigger is not set
  Framing is ESF, FDL is ansi, Line Code is B8ZS, Clock Source is Line.
  Data in current interval (742 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
  Total Data (last 73 15 minute intervals):
     1278491 Line Code Violations, 3 Path Code Violations,
     0 Slip Secs, 1 Fr Loss Secs, 177 Line Err Secs, 0 Degraded Mins,
     3 Errored Secs, 0 Bursty Err Secs, 1 Severely Err Secs, 227 Unavail Secs
.
.
.
```

Configuring Multilink Point-to-Point Protocol (Hardware-based)

Multilink Point to Point Protocol (MLPPP) allows you to combine interfaces which correspind to an entire T1 or E1 multilink bundle. You choose the number of bundles and the number of T1 or E1 lines in each bundle.

MLPPP Configuration Guidelines

The required conditions are:

- Only T1 or E1 links in a bundle

- All links on the same SPA
- Maximum of 12 links in a bundle.

Note Some notes about hardware-based MLPPP:

Only 3 fragmentation sizes are possible 128, 256 and 512 bytes

Fragmentation is enabled by default, default size is 512 bytes

Fragmentation size is configured using the **ppp multilink fragment-delay** command after using the **interface multilink** command. The least of the fragmentation sizes (among the 3 sizes possible) satisfying the delay criteria is configured. (e.g., a 192 byte packet causes a delay of 1 millisecond on a T1 link, so the nearest fragmentation size is 128 bytes.

The **show ppp multilink** command will indicate the MLPPP type and the fragmentation size:

```
Router# show ppp multilink
Multilink1, bundle name is Patriot2
Bundle up for 00:00:13
Bundle is Distributed
0 lost fragments, 0 reordered, 0 unassigned
0 discarded, 0 lost received, 206/255 load
0x0 received sequence, 0x0 sent sequence
Member links: 2 active, 0 inactive (max not set, min not set)
Se4/2/0/1:0, since 00:00:13, no frags rcvd
Se4/2/0/2:0, since 00:00:10, no frags rcvd
Distributed fragmentation on. Fragment size 512.  Multilink in Hardware.
```

Fragmentation is disabled explicitly by using the **no ppp multilink fragmentation** command after using the **interface multilink** command.

Create a Multilink Bundle

To create a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** *group-number*	Creates a multilink interface and enter multilink interface mode. • *group-number*—The group number for the multilink bundle.
Router(config-if)# **ip address** *address mask*	Sets the IP address for the multilink group. • *address*—The IP address. • *mask*—The IP netmask.

Assign an interface to a Multilink Bundle

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** For addressing information, refer to the "Interface Naming" section on page 20-2.	Selects the interface to configure and enters interface configuration mode.
Router(config-if)# **encapsulation ppp**	Enables PPP encapsulation.
Router(config-if)# **multilink-group** *group-number*	Assigns the interface to a multilink bundle. • *group-numbe*r—The multilink group number for the T1 or E1 bundle.
Router(config-if)# **ppp multilink**	Enables multilink PPP on the interface.
Repeat these commands for each interface you want to assign to the multilink bundle.	

Configuring fragmentation size on an MLPPP Bundle (optional)

To configure the fragmentation size on a multilink ppp bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** For addressing information, refer to the "Interface Naming" section on page 20-2.	Creates a multilink interface and enters multilink interface mode. • group-number—The group number for the multilink bundle. Range 1-2147483647
Router(config-if)# **ppp multilink fragment-delay** *delay*	Sets the fragmentation size satisfying the configured delay on the multilink bundle. • delay—delay in milliseconds

Disabling the fragmentation on an MLPPP Bundle (optional)

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface multilink** *group-number*	Creates a multilink interface and enters multilink interface mode. • group-number—The group number for the multilink bundle. Range 1-2147483647
Router(config-if)# **no pppp multilink fragmentation**	Disables the fragmentation on the multilink bundle.

Configuring MLFR

Multilink Frame Relay (MLFR) allows you to combine T1/E1 lines into a bundle that has the combined bandwidth of multiple T1/E1 lines. You choose the number of bundles and the number of T1/E1 lines in each bundle. This allows you to increase the bandwidth of your network links beyond that of a single T1/E1 line.

MLFR Configuration Guidelines

MLFR will function in hardware if all of the following conditions are met:

- Only T1 or E1 member links
- All links are on the same SPA
- Maximum of 12 links in a bundle

Create a Multilink Bundle

To create a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface mfr** *number*	Configures a multilink Frame Relay bundle interface. • *number*—The number for the Frame Relay bundle.
Router(config-if)# **frame-relay multilink bid** *name*	(Optional) Assigns a bundle identification name to a multilink Frame Relay bundle. • *name*—The name for the Frame Relay bundle. **Note** The bundle identification (BID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shut** and **no shut** commands in interface configuration mode.

Assign an Interface to a Multilink Bundle

To assign an interface to a multilink bundle, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** For addressing information, refer to the "Interface Naming" section on page 20-2.	Selects the interface to assign.

Command	Purpose
Router(config-if)# **encapsulation frame-relay mfr** *number* [*name*]	Creates a multilink Frame Relay bundle link and associates the link with a bundle. • *number*—The number for the Frame Relay bundle. • *name*—The name for the Frame Relay bundle.
Router(config-if)# **frame-relay multilink lid** *name*	(Optional) Assigns a bundle link identification name with a multilink Frame Relay bundle link. • *name*—The name for the Frame Relay bundle. **Note** The bundle link identification (LID) will not go into effect until the interface has gone from the down state to the up state. One way to bring the interface down and back up again is by using the **shut** and **no shut** commands in interface configuration mode.
Router(config-if)# **frame-relay multilink hello** *seconds*	(Optional) Configures the interval at which a bundle link will send out hello messages. The default value is 10 seconds. • *seconds*—Number of seconds between hello messages sent out over the multilink bundle.
Router(config-if)# **frame-relay multilink ack** *seconds*	(Optional) Configures the number of seconds that a bundle link will wait for a hello message acknowledgment before resending the hello message. The default value is 4 seconds. • *seconds*—Number of seconds a bundle link will wait for a hello message acknowledgment before resending the hello message.
Router(config-if)# **frame-relay multilink retry** *number*	(Optional) Configures the maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment. The default value is 2 tries. • *number*—Maximum number of times a bundle link will resend a hello message while waiting for an acknowledgment.

Verifying Multilink Frame Relay

Use the **show frame-relay multilink detailed** command to verify the Frame Relay multilinks:

```
router# show frame-relay multilink detailed

Bundle: MFR49, State = down, class = A, fragmentation disabled
 BID = MFR49
 No. of bundle links = 1, Peer's bundle-id =
 Bundle links:

  Serial6/0/0:0, HW state = up, link state = Add_sent, LID = test
    Cause code = none, Ack timer = 4, Hello timer = 10,
    Max retry count = 2, Current count = 0,
```

```
Peer LID = , RTT = 0 ms
Statistics:
Add_link sent = 21, Add_link rcv'd = 0,
Add_link ack sent = 0, Add_link ack rcv'd = 0,
Add_link rej sent = 0, Add_link rej rcv'd = 0,
Remove_link sent = 0, Remove_link rcv'd = 0,
Remove_link_ack sent = 0, Remove_link_ack rcv'd = 0,
Hello sent = 0, Hello rcv'd = 0,
Hello_ack sent = 0, Hello_ack rcv'd = 0,
outgoing pak dropped = 0, incoming pak dropped = 0
```

Invert Data on the T1/E1 Interface

If the interface on the 1-Port Channelized OC-3/STM-1 SPA is used to drive a dedicated T1 line that does not have B8ZS encoding, you must invert the data stream on the connecting CSU/DSU or on the interface. Be careful not to invert data on both the CSU/DSU and the interface, as two data inversions will cancel each other out. To invert data on a T1/E1 interface, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# **interface serial** For addressing information, refer to the "Interface Naming" section on page 20-2.	Selects the serial interface.
Router(config-if)# **invert data**	Inverts the data stream.

Use the **show running configuration** command to verify that invert data has been set:

```
router# show running configuration
.
.
.
interface Serial6/0/0:0
 no ip address
 encapsulation ppp
 logging event link-status
 load-interval 30
 invert data
 no cdp enable
 ppp chap hostname group1
 ppp multilink
 multilink-group 1
!
.
.
.
```

Configuring Multipoint Bridging

Multipoint bridging (MPB) enables the connection of multiple ATM PVCs, Frame Relay PVCs, BCP ports, and WAN Gigabit Ethernet subinterfaces into a single broadcast domain (virtual LAN), together with the LAN ports on that VLAN. This enables service providers to add support for Ethernet-based Layer 2 services to the proven technology of their existing ATM and Frame Relay legacy networks.

Customers can then use their current VLAN-based networks over the ATM or Frame Relay cloud. This also allows service providers to gradually update their core networks to the latest Gigabit Ethernet optical technologies, while still supporting their existing customer base.

For MPB configuration guidelines and restrictions and feature compatibility tables, see the "Configuring Multipoint Bridging" section on page 4-23 of Chapter 4, "Configuring the SIPs and SSC."

Configuring Bridging Control Protocol Support

The Bridging Control Protocol (BCP) enables forwarding of Ethernet frames over SONET networks and provides a high-speed extension of enterprise LAN backbone traffic through a metropolitan area. The implementation of BCP on the SPAs includes support for IEEE 802.1D, IEEE 802.1Q Virtual LAN (VLAN), and high-speed switched LANs.

For BCP configuration guidelines and restrictions and feature compatibility tables, see the "Configuring PPP Bridging Control Protocol Support" section on page 4-35 of Chapter 4, "Configuring the SIPs and SSC."

Changing a Channel Group Configuration

To alter the configuration of an existing channel group, the channel group needs to be removed first using the **no** form of the **channel-group** command. To remove an existing channel group, use the following commands:

Command	Purpose
Router# **configure terminal**	Enters global configuration mode.
Router(config)# For addressing information, refer to the "Interface Naming" section on page 20-2.	Select the controller to configure and enter controller configuration mode.
Router(config-controller)# **no channel-group** *t1 t1-number*	Select the channel group you want to remove. • *t1 t1-number*— channel-group number.

FRF.12 Guidelines

FRF.12 functions in hardware. Note the following:

- The fragmentation is configured at the main interface
- Only 3 fragmentation sizes are available - 128 bytes, 256 bytes, and 512 bytes.

LFI Guidelines

LFI can function two ways - using FRF.12 or MLPPP. MLPPP LFI can be done in both hardware and software while FRF.12 LFI is done only in hardware.

HW MLPPP LFI Guidelines

LFI using MLPPP will function only in hardware if there is just one member link in the MLPPP bundle. The link can be a fractional T1 or full T1. Note the following:

- The **ppp multilink interleave** command needs to be configured to enable interleaving.
- Only three fragmentation sizes are supported - 128 bytes, 256 bytes, and 512 bytes.
- Fragmentation is enabled by default, the default size being 512 bytes.
- A policy-map having a priority class needs to applied to main interface.

FRF.12 LFI Guidelines

LFI using FRF.12 is always done is hardware. Note the following:

- The fragmentation is configured at the main interface
- Only 3 fragmentation sizes are available - 128 bytes, 256 bytes, and 512 bytes.
- A policy-map having a priority class needs to applied to main interface.

Configuring QoS Features on Serial SPAs

The SIPs and SPAs support many QoS features using modular QoS CLI (MQC) configuration. For information about the QoS features supported by the serial SPAs, see the "Configuring QoS Features on a SIP" section on page 4-61 of Chapter 4, "Configuring the SIPs and SSC."

Saving the Configuration

To save your running configuration to nonvolatile random-access memory (NVRAM), use the following command in privileged EXEC configuration mode:

Command	Purpose
Router# **copy running-config startup-config**	Writes the new configuration to NVRAM.

For more information about managing configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2* and *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* publications.

Verifying the Interface Configuration

Besides using the **show running-configuration** command to display your Cisco 7600 series router configuration settings, you can use the **show interface serial** and the **show controllers serial** commands to get detailed information on a per-port basis for your1-Port Channelized OC-3/STM-1 SPA.

Verifying Per-Port Interface Status

To find detailed interface information on a per-port basis for the 1-Port Channelized OC-3/STM-1 SPA use the **show interface serial** command.

The following example provides sample output for interface port 0 on the SPA located in the first subslot of the Cisco 7600 SIP-200 installed in slot 2 of a Cisco 7600 series router:

```
Router# show interface serial 2/1/0.2/1:0
```

```
Serial2/1/0.2/1:0 is down, line protocol is down
Hardware is Channelized-T3
MTU 1500 bytes, BW 1536 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, crc 16, loopback not set
Keepalive set (10 sec)
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Available Bandwidth 1536 kilobits/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 2 interface resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions alarm present
VC 5: timeslot(s): 1-24, Transmitter delay 0, non-inverted data
UUT#sh int Serial2/1/0.3
Serial2/1/0.3 is down, line protocol is down
Hardware is CHOCx SPA
MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec, rely 255/255, load 1/255
Encapsulation HDLC, crc 16, loopback not set
Keepalive set (10 sec)
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Available Bandwidth 44210 kilobits/sec
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 parity

(Remaining output omitted)
```

Configuration Tasks

This section describes common configurations for the 1-Port Channelized OC-3/STM-1 SPA on a Cisco 7600 series router. It contains procedures for the following configurations:

- Configuring CRTP, page 20-22

Configuring CRTP

For information on configuring cRTP, see *Configuring Distributed Compressed Real-Time Protocol* at the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a00800b75cd.html

CHAPTER 21

Troubleshooting the Serial SPAs

This chapter describes techniques that you can use to troubleshoot the operation of your serial SPAs.

It includes the following sections:

The first section provides information about basic interface troubleshooting. If you are having a problem with your SPA, use the steps in the "General Troubleshooting Information" section on page 21-1 section to begin your investigation of a possible interface configuration problem.

To perform more advanced troubleshooting, see the other sections in this chapter.

For more information about troubleshooting serial lines, see the *Internetwork Troubleshooting Handbook* at http://www.cisco.com/univercd/cc/td/doc/cisintwk/itg_v1/index.htm.

General Troubleshooting Information

This section describes general information for troubleshooting SIPs and SPAs. It includes the following sections:

Interpreting Console Error Messages

To view the explanations and recommended actions for Cisco 7600 series router error messages, including messages related to Cisco 7600 series router SIPs and SPAs, refer to the following document:

- *Cisco 7600 Series Cisco IOS System Message Guide, 12.2SR*
- *System Error Messages for Cisco IOS Release 12.2S* (for error messages in Release 12.2S)

System error messages are organized in the documentation according to the particular system facility that produces the messages. The SIP and SPA error messages use the following facility names:

- Cisco 7600 SIP-200—C7600_SIP200
- 2-Port and 4-Port Channelized T3 SPA—SPA_CHOC_DSX

Using debug Commands

Along with the other **debug** commands supported on the Cisco 7600 series router, you can obtain specific debug information for SPAs on the Cisco 7600 series router using the **debug hw-module subslot** privileged EXEC command.

The **debug hw-module subslot** command is intended for use by Cisco Systems technical support personnel. For more information about the **debug hw-module subslot** command, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*.

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

For information about other **debug** commands supported on the Cisco 7600 series router, refer to the *Cisco IOS Debug Command Reference, Release 12.2* and any related feature documents for Cisco IOS Release 12.2 SX.

Using show Commands

There are several **show** commands that you can use to monitor and troubleshoot the SIPs and SPAs on the Cisco 7600 series router. This chapter describes using the **show interfaces** and **show controllers** commands to perform troubleshooting of your SPA.

For more information about **show** commands to verify and monitor SIPs and SPAs, see the following chapters of this guide:

- Chapter 18, "Configuring the 2-Port and 4-Port Clear Channel T3/E3 SPAs"
- Chapter 17, "Configuring the 8-Port Channelized T1/E1 SPA"
- Chapter 19, "Configuring the 2-Port and 4-Port Channelized T3 SPAs"

Performing Basic Interface Troubleshooting

You can perform most of the basic interface troubleshooting using the **show interfaces serial** command and examining several areas of the output to determine how the interface is operating.

The output of the **show interfaces serial** EXEC command displays information specific to serial interfaces.

Note The output of the **show interfaces serial** command will vary depending on the type of serial SPA.

This section describes how to use the **show interfaces serial** command to diagnose serial line connectivity problems in a wide-area network (WAN) environment. The following sections describe some of the important fields of the command output:

- Serial Lines: show interfaces serial Status Line Conditions, page 21-3
- Serial Lines: Increasing Output Drops on Serial Link, page 21-7
- Serial Lines: Increasing Input Drops on Serial Link, page 21-8
- Serial Lines: Increasing Input Errors in Excess of 1 Percent of Total Interface Traffic, page 21-9
- Serial Lines: Troubleshooting Serial Line Input Errors, page 21-9
- Serial Lines: Increasing Interface Resets on Serial Link, page 21-12
- Serial Lines: Increasing Carrier Transitions Count on Serial Link, page 21-13

Serial Lines: show interfaces serial Status Line Conditions

You can identify five possible problem states in the interface status line of the **show interfaces serial** display:

- Serial *x* is down, line protocol is down
- Serial *x* is up, line protocol is down
- Serial *x* is up, line protocol is up (looped)
- Serial *x* is up, line protocol is down (disabled)
- Serial *x* is administratively down, line protocol is down

The following example shows the interface statistics on the first port of a T3/E3 SPA installed in subslot 0 of the SIP located in chassis slot 5.

```
Router# show interfaces serial

Serial5/0/0 is up, line protocol is up
  Hardware is SPA-4T3E3
  Internet address is 110.1.1.2/24
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 234/255, rxload 234/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Last input 00:00:05, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 40685000 bits/sec, 115624 packets/sec
  5 minute output rate 40685000 bits/sec, 115627 packets/sec
     4653081241 packets input, 204735493724 bytes, 0 no buffer
     Received 4044 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     4652915555 packets output, 204728203520 bytes, 0 underruns
     0 output errors, 0 applique, 4 interface resets
```

```
0 output buffer failures, 0 output buffers swapped out
2 carrier transitions
```

Table 21-1 shows the interface status conditions, possible problems associated with the conditions, and solutions to those problems.

Table 21-1 Serial Lines: show interfaces serial Status Line Conditions

Status Line Condition	Possible Problem	Solution
Serial *x* is up, line protocol is up	—	This is the proper status line condition. No action is required.
Serial *x* is down, line protocol is down	The router is not sensing a carrier detect (CD) signal (that is, the CD is not active). The line is down or is not connected on the far end. Cabling is faulty or incorrect. Hardware failure has occurred in the channel service unit/data service uint (CSU/DSU).	**1.** Check the CD LEDs to see whether the CD is active, or insert a breakout box on the line to check for the CD signal. **2.** Verify that you are using the proper cable (see your hardware installation documentation). **3.** Insert a breakout box and check all control leads. **4.** Contact your leased-line or other carrier service to see whether there is a problem. **5.** Swap faulty parts. **6.** If you suspect faulty router hardware, change the serial line to another port. If the connection comes up, the previously connected interface has a problem.

Table 21-1 Serial Lines: show interfaces serial Status Line Conditions (continued)

Status Line Condition	Possible Problem	Solution
Serial *x* is up, line protocol is down	A local or remote router is misconfigured. Keepalives are not being sent by the remote router. A leased-line or other carrier service problem has occurred (noisy line or misconfigured or failed switch). A timing problem has occurred on the cable. A local or remote CSU/DSU has failed. Router hardware (local or remote) has failed.	1. Put the line in local loopback mode and use the **show interfaces serial** command to determine whether the line protocol comes up. **Note** If the line protocol comes up, a failed remote device is the likely problem. This solution will only work with High-Level Data Link Control (HDLC) encapsulation. For Frame Relay (FR) and Point-to-Point Protocol (PPP) encapsulation a looped interface will always have the line protocol down. In addition, you may need to change the encapsulation to HDLC to debug this issues. 2. If the problem appears to be on the remote end, repeat Step 1 on the remote interface. 3. Verify all cabling. Make certain that the cable is attached to the correct interface, the correct CSU/DSU, and the correct remote termination point. 4. Enable the **debug serial interface** EXEC command. **Note** First enable per interface debugging using the command "**debug interface serial** *x*", and depending on the encapsulation, enable the corresponding debug. • For HDLC: **debug serial interface** For PPP: **debug ppp negotiation** For FR: **debug frame-relay lmi** **Caution** Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Table 21-1 Serial Lines: show interfaces serial Status Line Conditions (continued)

Status Line Condition	Possible Problem	Solution
		5. If the line protocol does not come up in local loopback mode, and if the output of the **debug serial interface** EXEC command shows that the keepalive counter is not incrementing, a router hardware problem is likely. Swap router interface hardware. 6. If the line protocol comes up and the keepalive counter increments, the problem is *not* in the local router. 7. If you suspect faulty router hardware, change the serial line to an unused port. If the connection comes up, the previously connected interface has a problem.
Serial *x* is up, line protocol is up (looped)	A loop exists in the circuit. The sequence number in the keepalive packet changes to a random number when a loop is initially detected. If the same random number is returned over the link, a loop exists.	1. Use the **show running-config** privileged EXEC command to look for any **loopback** interface configuration command entries. 2. If you find a **loopback** interface configuration command entry, use the **no loopback** interface configuration command to remove the loop. 3. If you do not find the **loopback** interface configuration command, examine the CSU/DSU to determine whether they are configured in manual loopback mode. If they are, disable manual loopback. 4. Reset the CSU or DSU, and inspect the line status. If the line protocol comes up, no other action is needed. 5. If the CSU or DSU is not configured in manual loopback mode, contact the leased-line or other carrier service for line troubleshooting assistance.

Table 21-1 Serial Lines: show interfaces serial Status Line Conditions (continued)

Status Line Condition	Possible Problem	Solution
Serial *x* is up, line protocol is down (disabled)	A high error rate has occurred due to a remote device problem. A CSU or DSU hardware problem has occurred. Router hardware (interface) is bad.	1. Troubleshoot the line with a serial analyzer and breakout box. Examine the output of **show controller T1** or **show controller T3** or **show controller serial x** depending on whether the SPA is a T1/E1, CT3, or T3/E3. 2. Loop CSU/DSU (DTE loop). If the problem continues, it is likely that there is a hardware problem. If the problem does not continue, it is likely that there is a telephone company problem. 3. Swap out bad hardware, as required (CSU, DSU, switch, local or remote router).
Serial *x* is administratively down, line protocol is down	The router configuration includes the **shutdown** interface configuration command. A duplicate IP address exists.	1. Check the router configuration for the **shutdown** command. 2. Use the **no shutdown** interface configuration command to remove the **shutdown** command. 3. Verify that there are no identical IP addresses using the **show running-config** privileged EXEC command or the **show interfaces** EXEC command. 4. If there are duplicate addresses, resolve the conflict by changing one of the IP addresses.

Serial Lines: Increasing Output Drops on Serial Link

Output drops appear in the output of the **show interfaces serial** command when the system is attempting to hand off a packet to a transmit buffer but no buffers are available.

Symptom: Increasing output drops on serial link

Table 21-2 outlines the possible problem that might cause this symptom and describes solutions to that problem.

Table 21-2 Serial Lines: Increasing Output Drops on Serial Link

Possible Problem	Solution
Input rate to serial interface exceeds bandwidth available on serial link	1. Minimize periodic broadcast traffic, such as routing and Service Advertising Protocol (SAP) updates, by using access lists or by other means. For example, to increase the delay between SAP updates, use the **ipx sap-interval** interface configuration command. 2. Increase the output hold queue size in small increments (for instance, 25 percent), using the **hold-queue out** interface configuration command. 3. Implement priority queuing on slower serial links by configuring priority lists. For information on configuring priority lists, see the Cisco IOS configuration guides and command references. **Note** Output drops are acceptable under certain conditions. For instance, if a link is known to be overused (with no way to remedy the situation), it is often considered more preferable to drop packets than to hold them. This is true for protocols that support flow control and can retransmit data (such as TCP/IP and Novell Internetwork Packet Exchange [IPX]). However, some protocols, such as DECnet and local-area transport, are sensitive to dropped packets and accommodate retransmission poorly, if at all.

Serial Lines: Increasing Input Drops on Serial Link

Input drops appear in the output of the **show interfaces serial** EXEC command when too many packets from that interface are still being processed in the system.

Symptom: Increasing number of input drops on serial link

Table 21-3 outlines the possible problem that might cause this symptom and describes solutions to that problem.

Table 21-3 Serial Lines: Increasing Input Drops on Serial Link

Possible Problem	Solution
Input rate exceeds the capacity of the router, or input queues exceed the size of output queues	**Note** Input drop problems are typically seen when traffic is being routed between faster interfaces (such as Ethernet, Token Ring, and Fiber Distributed Data Interface [FDDI]) and serial interfaces. When traffic is light, there is no problem. As traffic rates increase, backups start occurring. Routers drop packets during these congested periods. 1. Increase the output queue size on common destination interfaces for the interface that is dropping packets. Use the **hold-queue** *number* **out** interface configuration command. Increase these queues by small increments (for instance, 25 percent) until you no longer see drops in the **show interfaces** command output. The default output hold queue limit is 40 packets. 2. Reduce the input queue size, using the **hold-queue** *number* **in** interface configuration command, to force input drops to become output drops. Output drops have less impact on the performance of the router than do input drops. The default input hold queue is 75 packets.

Serial Lines: Increasing Input Errors in Excess of 1 Percent of Total Interface Traffic

If input errors appear in the **show interfaces serial** command output, there are several possible sources of those errors. The most likely sources, along with possible solutions, are summarized in Table 21-4.

> **Note** Any input error value for cyclic redundancy check (CRC) errors, framing errors, or aborts above 1 percent of the total interface traffic suggests some kind of link problem that should be isolated and repaired.

Symptom: Increasing number of input errors in excess of 1 percent of total interface traffic.

Table 21-4 Serial Lines: Increasing Input Errors in Excess of 1 Percent of Total Interface Traffic

Possible Problem	Solution
The following problems can result in this symptom: • Faulty telephone company equipment • Noisy serial line • Incorrect clocking configuration • Incorrect cable or cable that is too long • Bad cable or connection • Bad CSU or DSU • Bad router hardware • Data converter or other device being used between router and DSU	**Note** Cisco strongly recommends against the use of data converters when you are connecting a router to a WAN or a serial network. 1. Use a serial analyzer to isolate the source of the input errors. If you detect errors, there likely is a hardware problem or a clock mismatch in a device that is external to the router. 2. Use the loopback and **ping** tests to isolate the specific problem source. 3. Look for patterns. For example, if errors occur at a consistent interval, they could be related to a periodic function, such as the sending of routing updates.

Serial Lines: Troubleshooting Serial Line Input Errors

Table 21-5 describes the various types of input errors displayed by the **show interfaces serial** command, possible problems that might be causing the errors, and solutions to those problems.

Table 21-5 Serial Lines: Troubleshooting Serial Line Input Errors

Input Error Type (Field Name)	Possible Problem	Solution
CRC errors (CRC)	CRC errors occur when the CRC calculation does not pass (indicating that data is corrupted) for one of the following reasons: • The serial line is noisy. • The serial cable is too long, or the cable from the CSU/DSU to the router is not shielded. • Serial clock transmit external (SCTE) mode is not enabled on DSU. • The CSU line clock is incorrectly configured. • A ones density problem has occurred on the T1 link (incorrect framing or coding specification).	1. Ensure that the line is clean enough for transmission requirements. Shield the cable, if necessary. 2. Make sure that the cable is within the recommended length (no more than 50 feet [15.24 meters], or 25 feet [7.62 meters] for a T1 link). 3. Ensure that all devices are properly configured for a common line clock. Set SCTE on the local and remote DSU. If your CSU/DSU does not support SCTE, see the section "Inverting the Transmit Clock," later in this chapter. 4. Make certain that the local and remote CSU/DSU are configured for the same framing and coding scheme as that used by the leased-line or other carrier service (for example, Extended Superframe Format/binary eight-zero substitution [ESF/B8ZS]). 5. Contact your leased-line or other carrier service, and have it perform integrity tests on the line.

Table 21-5 Serial Lines: Troubleshooting Serial Line Input Errors (continued)

Input Error Type (Field Name)	Possible Problem	Solution
Framing errors (frame)	A framing error occurs when a packet does not end on an 8-bit byte boundary for one of the following reasons: • The serial line is noisy. • The cable is improperly designed, the serial cable is too long, or the cable from the CSU or DSU to the router is not shielded. • SCTE mode is not enabled on the DSU, the CSU line clock is incorrectly configured, or one of the clocks is configured for local clocking. • A ones density problem has occurred on the T1 link (incorrect framing or coding specification).	1. Ensure that the line is clean enough for transmission requirements. Shield the cable, if necessary. Make certain that you are using the correct cable. 2. Make sure that the cable is within the recommended length (no more than 50 feet [15.24 meters], or 25 feet [7.62 meters] for a T1 link). 3. Ensure that all devices are properly configured to use a common line clock. Set SCTE on the local and remote DSU. 4. Make certain that the local and remote CSU/DSU is configured for the same framing and coding scheme as that used by the leased-line or other carrier service (for example, ESF/B8ZS). 5. Contact your leased-line or other carrier service, and have it perform integrity tests on the line.

Table 21-5 Serial Lines: Troubleshooting Serial Line Input Errors (continued)

Input Error Type (Field Name)	Possible Problem	Solution
Aborted transmission (abort)	Aborts indicate an illegal sequence of 1 bit (more than seven in a row). The following are possible reasons for this to occur: • SCTE mode is not enabled on DSU. • The CSU line clock is incorrectly configured. • The serial cable is too long, or the cable from the CSU or DSU to the router is not shielded. • A ones density problem has occurred on the T1 link (incorrect framing or coding specification). • A packet was terminated in the middle of transmission (typical cause is an interface reset or a framing error or a buffer overrun). • A hardware problem has occurred (bad circuit, bad CSU/DSU, or bad sending interface on remote router).	1. Ensure that all devices are properly configured to use a common line clock. Set SCTE on the local and remote DSU. 2. Shield the cable, if necessary. Make certain that the cable is within the recommended length (no more than 50 feet [15.24 meters], or 25 feet [7.62 meters] for a T1 link). Ensure that all connections are good. 3. Check the hardware at both ends of the link. Swap faulty equipment, as necessary. 4. Lower data rates and determine whether aborts decrease. 5. Use local and remote loopback tests to determine where aborts are occurring. 6. Contact your leased-line or other carrier service, and have it perform integrity tests on the line.

Serial Lines: Increasing Interface Resets on Serial Link

Interface resets that appear in the output of the **show interfaces serial** EXEC command are the result of missed keepalive packets.

Symptom: Increasing interface resets on serial link

Table 21-6 outlines the possible problems that might cause this symptom and describes solutions to those problems.

Table 21-6 Serial Lines: Increasing Interface Resets on Serial Link

Possible Problem	Solution
The following problems can result in this symptom: • Congestion on link (typically associated with output drops) • Bad line causing CD transitions • Possible hardware problem at the CSU, DSU, or switch	When interface resets are occurring, examine other fields of the **show interfaces serial** command output to determine the source of the problem. Assuming that an increase in interface resets is being recorded, examine the following fields: 1. If there is a high number of output drops in the **show interfaces serial** output, see the "Serial Lines: Increasing Output Drops on Serial Link" section on page 21-7. 2. Check the Carrier Transitions field in the **show interfaces serial** command display. If carrier transitions are high while interface resets are being registered, the problem is likely to be a bad link or a bad CSU or DSU. Contact your leased-line or carrier service, and swap faulty equipment, as necessary. 3. Examine the Input Errors field in the **show interfaces serial** command display. If input errors are high while interface resets are increasing, the problem is probably a bad link or a bad CSU/DSU. Contact your leased-line or other carrier service, and swap faulty equipment, as necessary.

Serial Lines: Increasing Carrier Transitions Count on Serial Link

Carrier transitions appear in the output of the **show interfaces serial** EXEC command whenever there is an interruption in the carrier signal (such as an interface reset at the remote end of a link).

Symptom: Increasing carrier transitions count on serial link

Table 21-7 outlines the possible problems that might cause this symptom and describes solutions to those problems.

Table 21-7 Serial Lines: Increasing Carrier Transitions Count on Serial Link

Possible Problem	Solution
The following problems can result in this symptom: • Line interruptions due to an external source (such as physical separation of cabling, red or yellow T1 alarms, or lightning striking somewhere along the network) • Faulty switch, DSU, or router hardware	1. Check hardware at both ends of the link (attach a breakout box or a serial analyzer, and test to determine the source of problems). 2. If an analyzer or breakout box is incapable of identifying any external problems, check the router hardware. 3. Swap faulty equipment, as necessary.

Using Bit Error Rate Tests

BER test circuitry is built into most of the serial SPAs. With BER tests, you can test cables and signal problems in the field. You can configure individual T1 lines to run BER tests, but only one BER test circuit exists for all 28 T1 lines. Hence, only one BER test can be run on a single T3 port at any given time.

There are two categories of test patterns that can be generated by the onboard BER test circuitry: pseudorandom and repetitive. Pseudorandom test patterns are exponential numbers and conform to the CCITT/ITU O.151 and O.153 specifications; repetitive test patterns are all zeros, all ones, or alternating zeros and ones.

A description of the test patterns follows:

- Pseudorandom test patterns:
  - 2^15 (per CCITT/ITU O.151)
  - 2^20 (per CCITT/ITU O.153)
  - 2^23 (per CCITT/ITU O.151)
  - QRSS (quasi-ramdom signal sequence) (per CCIT/ITU O.151)
- Repetitive test patterns:
  - All zeros (0s)
  - All ones (1s)
  - Alternating zeros (0s) and ones (1s)

Additional patterns are available as of Cisco IOS Release 12.2(SRC) on the 1-Port Channelized OC3/STM-1 and 2- and 4-Port Channelized T3 SPAs:

- 1-in-8—1-in-8 test pattern
- 3-in-24—3-in 24 test pattern
- 2^15-inverted—2^15-1 (inverted) O.151 test pattern
- 2^23-inverted—2^23-1 (inverted) O.151 test pattern
- 2^9—2^9-1 test pattern
- 2^11—2^11-1 test pattern
- 2^20-O153—2^20-1 O.153 test pattern
- 2^20-QRSS—2^20-1 QRSS O.151 test pattern
- 55Octet—55 Octet pattern
- 55Daly—55 Octet Daly pattern
- DS0-1, DS0-2, DS0-3, DS0-4—DS0 1, DS0 2, DS0 3, DS0 4 test patterns

Both the total number of error bits received and the total number of bits received are available for analysis. You can set the testing period from 1 minute to 14,400 minutes (240 hours), and you can also retrieve the error statistics anytime during the BER test.

When running a BER test, your system expects to receive the same pattern that it is transmitting. To help ensure this:

- Use a loopback at a location of your choice in the link or network. To see how to configure a loopback, go to the "Using loopback Commands" section on page 21-16.
- Configure remote testing equipment to transmit the same BER test pattern at the same time.

Configuring a BER Test

To send a BER test pattern on an interface, see the **bert pattern** command description in the Cisco IOS Release 12.2SR command reference documents.

Viewing a BER Test

You can view the results of a BER test with the **show controllers** command.

You can view the results of a BER test at the following times:

- After you terminate the test using the **no bert** command.
- After the test runs completely.
- Anytime during the test (in real time).

```
Router# show controllers serial T3 1/0/0
T3 1/0/0 is up.
C2T3 H/W Version : 3, C2T3 ROM Version : 0.79, C2T3 F/W Version : 0.29.0
T3 1/0/0 T1 1
No alarms detected.
Clock Source is internal.
BERT test result (running)
   Test Pattern : 2^15, Status : Sync, Sync Detected : 1
   Interval : 5 minute(s), Time Remain : 5 minute(s)
   Bit Errors(Since BERT Started): 6 bits,
   Bits Received(Since BERT start): 8113 Kbits
   Bit Errors(Since last sync): 6 bits
   Bits Received(Since last sync): 8113 Kbits
```

Interpreting BER Test Results

Table 21-8 explains the output of the preceding command.

Table 21-8 Interpreting BER Test Results

Field	Description
BERT test result (running)	Indicates the current state of the test. In this case, “running” indicates that the BER test is still in progress. After a test is completed, “done” is displayed.
Test Pattern : 2^15, Status : Sync, Sync Detected : 1	Indicates the test pattern you selected for the test (2^15), the current synchronization state (sync), and the number of times synchronization has been detected during this test (1).

Table 21-8 Interpreting BER Test Results (continued)

Field	Description
Interval : 5 minute(s), Time Remain : 5 minute(s)	Indicates the time the test takes to run and the time remaining for the test to run. If you terminate a BER test, you receive a message similar to the following: `Interval : 5 minute(s), Time Remain : 2 minute(s) (unable to complete)` "Interval: 5 minutes" indicates the configured run time for the test. "Time Remain : 2 minutes" indicates the time remaining in the test prior to termination. "(Unable to complete)" signifies that you interrupted the test.
Bit Errors(Since BERT Started): 6 bits Bits Received(Since BERT start): 8113 Kbits Bit Errors(Since last sync): 6 bits Bits Received(Since last sync): 8113 Kbits	These four lines show the bit errors that have been detected versus the total number of test bits that have been received since the test started and since the last synchronization was detected.

Using loopback Commands

Loopback support is useful for testing the interface without connectivity to the network, or for diagnosing equipment malfunctions between the interface and a device. The 2-Port and 4-Port Clear Channel T3/E3 SPA supports both an internal and an external loopback mode. The external loopback mode requires the use of a loopback cable and implements a loopback through the transceiver on the SPA.

You can also configure an internal loopback without the use of a loopback cable that implements a loopback at the PHY device internally. By default, loopback is disabled.

To configure local loopback, use the following commands:

Command	Purpose
`Router# configure terminal`	Enters global configuration mode.

T3/E3
`Router(config)# interface serial slot/subslot/port`
T1/E1
`Router(config)# controller slot/subslot/port`
	Selects the interface to configure.
	• *slot/subslot/port*—Specifies the location of the interface.
	• *slot/subslot/port*—Specifies the location of the T1/E1 controller.

T3/E3
`Router(config-if)# loopback {local | dte | network {line | payload} | remote}`
T1/E1
`Router(config-controller)# loopback {local [line | payload] | diag}`
	Specifies the loopback mode.
	• **local**—Loop back after going through the framer toward the terminal.
	• **dte**—Loop back after the LIU towards the terminal.
	• **network**—Loop back towards the network.
	• **remote**—Send FEAC to set remote in loopback.
	• **line**—Loop back toward network before going through framer.
	• **payload**—Loop back toward network after going through framer.
	• **diag**—Loop back after going through the HDLC controller towards the terminal.

Verifying Loopback Mode

```
Router# show interfaces serial 6/0/0
Serial6/0/0 is up, line protocol is up (looped)
  Hardware is SPA-4T3E3
  MTU 4470 bytes, BW 44210 Kbit, DLY 200 usec,
     reliability 255/255, txload 225/255, rxload 221/255
  Encapsulation FRAME-RELAY, crc 16, loopback set (local)
  Keepalive set (10 sec)
  LMI enq sent  13281, LMI stat recvd 13280, LMI upd recvd 0, DTE LMI up
  LMI enq recvd 1, LMI stat sent  0, LMI upd sent  0
  LMI DLCI 1023  LMI type is CISCO  frame relay DTE
  FR SVC disabled, LAPF state down
  Broadcast queue 0/256, broadcasts sent/dropped 0/0, interface broadcasts 0
  Last input 00:00:07, output 00:00:00, output hang never
  Last clearing of "show interface" counters 1d12h
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 612756
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 38446000 bits/sec, 109217 packets/sec
  5 minute output rate 39023000 bits/sec, 110854 packets/sec
     14601577951 packets input, 642478074437 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicast)
     0 runts, 0 giants, 0 throttles
              0 parity
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
```

```
  14545044296 packets output, 639982568049 bytes, 0 underruns
  0 output errors, 0 applique, 1 interface resets
  0 output buffer failures, 0 output buffers swapped out
  0 carrier transitions
rxLOS inactive, rxLOF inactive, rxAIS inactive
txAIS inactive, rxRAI inactive, txRAI inactive
```

Using the Cisco IOS Event Tracer to Troubleshoot Problems

Note This feature is intended for use as a software diagnostic tool and should be configured only under the direction of a Cisco Technical Assistance Center (TAC) representative.

The Event Tracer feature provides a binary trace facility for troubleshooting Cisco IOS software. This feature gives Cisco service representatives additional insight into the operation of the Cisco IOS software and can be useful in helping to diagnose problems in the unlikely event of an operating system malfunction or, in the case of redundant systems, Route Processor switchover.

Event tracing works by reading informational messages from specific Cisco IOS software subsystem components that have been preprogrammed to work with event tracing, and by logging messages from those components into system memory. Trace messages stored in memory can be displayed on the screen or saved to a file for later analysis.

The SPAs currently support the “spa” component to trace SPA OIR-related events.

For more information about using the Event Tracer feature, refer to the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1829/products_feature_guide09186a0080087164.html

Preparing for Online Insertion and Removal of a SPA

The Cisco 7600 series router supports online insertion and removal (OIR) of the SIP, in addition to each of the SPAs. Therefore, you can remove a SIP with its SPAs still intact, or you can remove a SPA independently from the SIP, leaving the SIP installed in the router.

This means that a SIP can remain installed in the router with one SPA remaining active, while you remove another SPA from one of the SIP subslots. If you are not planning to immediately replace a SPA into the SIP, then be sure to install a blank filler plate in the subslot. The SIP should always be fully installed with either functional SPAs or blank filler plates.

For more information about activating and deactivating SPAs in preparation for OIR, see the “Preparing for Online Insertion and Removal of SIPs and SPAs” topic in the “Troubleshooting a SIP” chapter in this guide.

PART 8

IPSec VPN Shared Port Adapter

CHAPTER 22

Overview of the IPSec VPN SPA

This chapter provides an overview of the release history, feature, and Management Information Base (MIB) support for the IPSec VPN SPA.

This chapter includes the following sections:

Release History

Release	Modification
Cisco IOS Release 12.2(18)SXF6	Support was introduced for the IPSec anti-replay window size feature in the SX release train.

Cisco IOS Release 12.2(33)SRA	Support was introduced for the following features: front door VRF, IPSec Virtual Tunnel Interface (VTI), Crypto-connect alternative (CCA) mode, certificate to ISAKMP profile mapping, Call Admission Control, periodic message option (now supported in Dead Peer Detection), Reverse Route Injection (RRI), IPSec anti-replay window size, IPSec preferred peer, local certificate storage location, optional OCSP nonces, persistent self-signed certificates, certificate chain verification, Easy VPN remote RSA signature storage, and IPSec and IKE MIB support for Cisco VRF-aware IPSec. Note the following changes from previous releases: • Tunnel capacity has been increased to 16,000 tunnels. • Support has been added for the following commands: – **clear crypto engine accelerator counter** command—To clear platform and network interface controller statistics. – **show crypto engine accelerator statistic** command—To display platform and network interface controller statistics. – **show crypto eli** command— To display how many IKE-SAs and IPSec sessions are active and how many Diffie-Hellman keys are in use for each IPSec VPN SPA. • Cisco IOS Release 12.2(33)SRA is only supported on Supervisor Engine 32 and Supervisor Engine 720. • Unlike previous releases, support is not included for IPSec stateful failover using HSRP and SSP. • The **crypto engine subslot** command has been replaced by the **crypto engine slot** command. • The one large configuration chapter has been restructured into seven smaller chapters, and a table has been added that describes release-dependent features. • The "Interoperability" section on page 22-9 has been expanded to include tables that differentiate Supervisor and line card support by release.
Cisco IOS Release 12.2(18)SXF2	Support was introduced for Supervisor Engine 2, Supervisor Engine 32, and the configuration of IP multicast over a GRE tunnel.
Cisco IOS Release 12.2(18)SXE5	Support was introduced for two new GRE takeover commands: • **crypto engine gre supervisor** command—To configure the router to process Generic Routing Encapsulation (GRE) using the Supervisor Engine hardware or the Route Processor (RP). • **crypto engine gre vpnblade** command—To configure the router to process Generic Routing Encapsulation (GRE) using the IPSec VPN SPA.
Cisco IOS Release 12.2(18)SXE2	Support for the IPSec VPN SPA was introduced on the Cisco 7600 SSC-400 on the Cisco 7600 series router.

Overview of the IPSec VPN SPA

The IPSec VPN SPA is a Gigabit Ethernet IP Security (IPSec) cryptographic SPA that you can install in a Cisco 7600 series router to provide hardware acceleration for IPSec encryption and decryption, generic routing encapsulation (GRE), and Internet Key Exchange (IKE) key generation. These services are used when you configure an IPSec virtual private network (VPN).

The existing implementation of IPSec in Cisco IOS supports the entire suite of security protocols including Authentication Header (AH), Encapsulating Security Payload (ESP), and IKE. The resources consumed by these activities are significant and make it difficult to achieve line-rate transmission speeds over secure VPNs. To address this problem, certain platforms with large VPN bandwidth requirements are now implementing bump-in-the-wire (BITW) IPSec hardware cards in conjunction with the hardware forwarding engines. These platforms off-load policy enforcement, as well as bulk encryption and forwarding, from the Route Processor (RP) so that it is not required to look at each packet coming through the router. This frees up resources that can be used for session establishment, key management, and other features. The IPSec VPN SPA provides a bump-in-the-wire (BITW) IPSec implementation using virtual LANs (VLANs) for a Cisco 7600 series router.

Note BITW is an IPSec implementation that starts egress packet processing after the IP stack has finished with the packet and completes ingress packet processing before the IP stack receives the packet.

The IPSec VPN SPA appears to the CLI as a SPA with two Gigabit Ethernet ports. The IPSec VPN SPA has no external connectors; the Gigabit Ethernet ports connect the IPSec VPN SPA to the router backplane and Switch Fabric Module (SFM) (if installed).

One Gigabit Ethernet port handles all the traffic going to and coming from the router outside ports. The other Gigabit Ethernet port handles all traffic going to and coming from the LAN or router inside ports. The IPSec VPN SPA can use multiple physical ports to connect to the outside through WAN routers. The physical ports may be attached to the IPSec VPN SPA through a VLAN called the port-VLAN (or pvlan).

Packets that are received from the WAN routers pass through the IPSec VPN SPA for IPSec processing. The packets are output on a dedicated VLAN called the interface or inside VLAN (or ivlan).

Depending on the configuration mode (VRF mode or crypto-connect mode), the ivlan or pvlan may be configured explicitly or may be allocated implicitly by the system.

On the LAN side, traffic between the LAN ports can be routed or bridged on multiple Fast Ethernet or Gigabit Ethernet ports. Because the LAN traffic is not encrypted or decrypted, it does not pass through the IPSec VPN SPA.

The IPSec VPN SPA does not maintain routing information, route, or change the MAC header of a packet (except for the VLAN ID from one VLAN to another).

Feature Table

The following table displays supported and unsupported features of the IPSec VPN SPA according to the software release:

Feature Name	Software Release 12.2					
	SXB	SXD	SXE	SXF	SRA	SRB
Global VRF	N	Y	Y	Y	Y	Y

Front-door VRF	N	N	N	N	Y	Y
GRE takeover always	Y	Y	N	N	N	N
Enhanced GRE takeover (if supervisor engine can't process)	N	N	Y	Y	Y	Y
Multicast over GRE	N	N	N	Y	Y	Y
Multicast over mGRE / DMVPN	N	N	N	N	N	N
Multicast Scalability Enhancement (single SPA mode)	N	N	N	Y	Y	Y
IPSec Static Virtual Tunnel Interface (sVTI)	N	N	N	N	Y	Y
Advanced Encryption Standard (AES) with SPA-IPSEC-2G only	N	N	SXE 2	Y	Y	Y
ISAKMP keyring	N	SXD 1	Y	Y	Y	Y
SafeNet Client support	Y	Y	Y	Y	Y	Y
Peer filtering (SafeNet Client support)	N	N	N	N	N	N
Certificate to ISAKMP profile mapping	N	SXD 1	Y	Y	Y	Y
Encrypted preshared key	Y	Y	Y	Y	Y	Y
IKE Aggressive Mode Initiation	N	N	N	N	N	N
Call Admission Control (CAC) for IKE	N	N	N	N	Y	Y
Dead Peer Detection (DPD) on-demand	Y	Y	Y	Y	Y	Y
Dead Peer Detection (DPD) periodic message option	N	N	N	N	Y	Y
IPSec Network Address Translation Transparency (NAT-T)	Y	Y	Y	Y	Y	Y
Look-Ahead Fragmentation (LAF) with IPSec	Y	Y	Y	Y	Y	Y
Reverse Route Injection (RRI)	Y	Y	Y	Y	Y	Y
IPSec anti-replay window size	N	N	N	Y	Y	Y
IPSec preferred peer	Y	Y	Y	Y	Y	Y
Per-crypto map (and global) IPSec security association (SA) idle timers	N	Y	Y	Y	Y	Y

Distinguished name-based crypto maps	X	Y	Y	Y	Y	Y
Sequenced Access Control Lists (ACLs)	N	Y	Y	Y	Y	Y
Deny policy configuration enhancements	N	Y	Y	Y	Y	Y
IPSec VPN SPA mls Quality of service trust (QoS) support	Y	Y	Y	Y	Y	Y
Multiple RSA key pair support	N	N	N	N	Y	Y
Protected private key storage	N	N	N	N	Y	Y
Trustpoint CLI	N	N	N	N	Y	Y
Query mode per trustpoint	N	N	N	N	N	N
Local certificate storage location	N	N	N	N	Y	Y
Direct HTTP enroll with CA servers	Y	Y	Y	Y	Y	Y
Manual certificate enrollment (TFTP and Cut-and-Paste)	N	N	N	N	Y	Y
Certificate autoenrollment	N	N	N	N	Y	Y
Key rollover for Certificate Authority (CA) renewal	N	N	N	N	N	N
Public-key infrastructure (PKI) query multiple servers	N	N	N	N	N	N
Online Certificate Status Protocol (OCSP)	N	N	N	N	N	N
Optional OCSP nonces	N	N	N	N	N	N
Certificate security attribute-based access control	N	N	N	N	N	N
PKI AAA authorization using entire subject name	N	N	N	N	N	N
PKI local authentication using subject name	N	N	N	N	Y	Y
Source interface selection for outgoing traffic with certificate authority	N	N	N	N	N	N
Persistent self-signed certificates as IOS CA server	N	N	N	N	N	N
Certificate chain verification	N	N	N	N	N	N
Multi-tier certitficate support	Y	Y	Y	Y	Y	Y

Dynamic Multipoint VPN Phase 2 (DMVPN) (mGRE; TP & NHRP)	N	N	SXE 2	Y	Y	Y
Easy VPN Server enhanced features	N	N	N	N	N	N
Easy VPN Server basic features	Y	Y	Y	Y	Y	Y
Interop with Easy VPN Remote	Y	Y	Y	Y	Y	Y
Interop with Easy VPN Remote RSA signature storage	N	N	N	N	Y	Y
Stateless failover using the Hot Standby Router Protocol (HSRP)	Y	Y	Y	Y	Y	Y
Stateful failover using HSRP and SSP	N	Y	Y	Y	N	N
Blade-to-Blade stateful failover	Y	Y	Y	Y	Y	Y
IPSec VPN Monitoring (IPSec Flow MIB)	Y	Y	Y	Y	Y	Y
IPSec VPN Accounting (start / stop / interim records)	N	Y	Y	Y	Y	Y
Crypto Conditional Debug support	N	N	N	Y	Y	Y
show crypto eli command	Y	Y	Y	Y	Y	Y
show crypto engine accelerator statistic command	N	N	N	N	Y	Y
clear crypto engine accelerator counter command	N	N	N	N	Y	Y
Policy Based Routing (PBR) on LAN interface	Y	Y	Y	Y	Y	Y
Policy Based Routing (PBR) on tunnel interface or interface VLAN	N	N	N	N	N	N
Blade-to-Blade failover (IPSec Stateful Failover) with DMVPN in VRF mode	N	N	N	N	N	N
Path MTU Discovery (PMTUD) is supported only in crypto connect mode	N	N	N	N	Y	Y
16K Tunnels (16K IKE & IPSec tunnels)	N	N	N	N	Y	Y
Switching between VRF and crypto-connect modes require reboot	N	Y	Y	Y	Y	Y
GRE keepalives on tunnel protection (TP) tunnels (per caveat CSCeb38231)	N	N	N	N	N	N

GRE keepalives on mGRE/DMVPN tunnels	N	N	N	N	N	N
A DMVPN hub router behind a NAT gateway	N	N	N	N	N	N
Multicast transit traffic over DMVPN tunnels	N	N	N	N	N	N
Non-IP version 4 (IPv4) traffic over TP tunnels	N	N	N	N	N	N
PCE (packet classification enhancement)(per caveat CSCeh34388)	N	N	N	N	N	N
DMVPN hierarchical hubs (DMVPN phase 3 feature)	N	N	N	N	N	N
Support for Supervisor Engine 2	Y	Y	Y	Y	N	N
Support for the VPNSM	Y	Y	Y	Y	N	N
Switching between hardware and software crypto modes	N	N	N	N	N	N
All serial interfaces with crypto connect mode must have "ip unnumber null 0"	N	N	N	N	N	Y
Manual key	N	N	N	Y	N	N
Tunnel Endpoint Discovery	Y	Y	Y	Y	N	N
Transport Adjacency Transform Set (AH and ESP together)	N	N	N	N	N	N
VPN SPA supported with VSS	N	N	N	N	N	N
WAN line card redundancy (HSRP)	N	N	N	N	N	N
Compression	N	N	N	N	N	N
Transit IPSec packets	N	N	N	N	N	N
Crypto Connect Mode						
Blade-to-Blade failover (IPSec Stateful Failover) with DMVPN	N	N	N	N	Y	Y
Point-to-Point GRE + Tunnel Protection and VTI	N	N	N	N	N	N
DMVPN	N	N	SXE 2	Y	Y	Y

IPSec VPNSPA mls QoS Trust on mGRE	N	N	N	N	N	N
Software crypto	Y	Y	Y	Y	N	N
IP header options through IPSec tunnels	N	N	N	N	N	N
VRF / CCA Mode						
Nested IPSec packets (transport adjacency only)	Y	Y	Y	Y	Y	Y
Software crypto	Y	Y	Y	Y	N	N
IP header options through IPSec tunnels	N	N	N	N	N	N
MPLS over GRE/IPSec (tag switching on tunnel interfaces)	N	N	N	N	N	N
PE-PE encryption (IPSec only) over MPLS	N	N	N	N	N	N
PE-PE encryption (Tunnel Protection) over MPLS (per caveat CSCef44310)	N	N	N	N	N	N
Software-switched TP tunnels (per caveat CSCef63742)	N	N	N	N	N	N
IPSec VPN SPA mls QoS on all tunnel interfaces: IPSec; GRE; mGRE	N	N	N	N	N	N
Per-VRF AAA with RADIUS	N	N	N	N	N	Y
Per-VRF AAA with TACACS	N	N	N	N	N	Y
Multicast over VTI	N	N	N	N	N	N
Ingress features (ACL, PBR, inbound service policy) on the outside interface	N	N	N	N	N	N
Outbound Service Policy on the outside interface	Y	Y	Y	Y	Y	Y
IPSec SA using crypto map created in Transport Mode	N	N	N	N	N	N

Recent changes

Significant recent feature changes include:

- Capacity
  - 16,000 tunnels (as of Cisco IOS Release 12.2(33)SRA)

Note Capacities are typically higher when IKE keepalive uses Dead Peer Detection (DPD).

- Configuration, management, and reporting
  - **show crypto eli** command (as of Cisco IOS Release 12.2(33)SRA)
  - **clear crypto engine accelerator counter** command (as of Cisco IOS Release 12.2(33)SRA)
  - **show crypto engine accelerator statistic** command (as of Cisco IOS Release 12.2(33)SRA)
  - **crypto engine gre supervisor** command (as of Cisco IOS Release 12.2(18)SXE5)
  - **crypto engine gre vpnblade** command (as of Cisco IOS Release 12.2(18)SXE5)
  - **crypto engine slot** command (as of Cisco IOS Release 12.2(33)SRA)

Note As of Cisco IOS Release 12.2(33)SRA, the **crypto engine subslot** command used in previous releases has been replaced with the **crypto engine slot** command (of the form **crypto engine slot** *slot/subslot* {**inside** | **outside**}). The **crypto engine subslot** command is no longer supported. When upgrading, ensure that this command has been modified in your start-up configuration to avoid extended maintenance time.

Interoperability

Supervisor Engine support varies based on the release.

Table 22-1 lists the supported Supervisor Engines for each release.

Table 22-1 Supervisor Engine Support for the IPSec VPN SPA by Release

Release	Supervisor Supported
Cisco IOS Release 12.2(33)SRA	Supervisor Engine 720 Supervisor Engine 32
Cisco IOS Release 12.2(18)SXF2	Supervisor Engine 720 Supervisor Engine 32 Supervisor Engine 2
Cisco IOS Release 12.2(18)SXE5	Supervisor Engine 720
Cisco IOS Release 12.2(18)SXE2	Supervisor Engine 720

Line card module support varies based on the release.

Table 22-2 lists the known supported line card modules for each release. Note the following guidelines when using this table:

- An “X” in the Qualified column indicates the module was tested; an “X” in the Supported column indicates that the module is supported.
- If the module has a footnote beside the “X” in the Supported column, although the module is supported, some restrictions apply. See the footnote below the table for details of the restriction.
- If the module has an “X” in the Supported column but not in the Qualified column, although the module was not specifically tested, it is supported.

Any line card modules not specifically listed in the table are not supported by TAC/BU.

Table 22-2 Line Card Module Support for the IPSec VPN SPA by Release

Line Card Module	Cisco IOS Release 12.2(18)SX		Cisco IOS Release 12.2(33)SR	
	Qualified	Supported	Qualified	Supported
7600-SIP-200 With the following SPAs: SPA-2XOC3-ATM= SPA-2XOC3-POS= SPA-2XT3/E3	X	X	X	X
7600-SIP-400 With the following SPAs: SPA-1XOC12-ATM= SPA-2X0C3-ATM= SPA-2X1GE		X[1]	X[2]	X
7600-SIP-600 With the following SPAs: SPA-1X10GE SPA-10X1GE			X[3]	X
7600-SSC-400	X	X	X	X
OSM-2OC48/1DPT-SI		X		X
OSM-2OC48/1DPT-SL		X		X
OSM-2OC48/1DPT-SS	X	X		X
OSM-8OC3-POS-MM	X	X	X	X
OSM-8OC3-POS-SI		X		X
OSM-8OC3-POS-SI+		X		X
OSM-8OC3-POS-SL		X		X
OSM-16OC3-POS-MM+	X	X	X	X
OSM-16OC3-POS-SI		X		X
OSM-16OC3-POS-SI+		X		X
OSM-16OC3-POS-SL		X		X
OSM-2+4GE-WAN+	X	X		X
WS-6182-2PA	X	X	X	X
WS-6582-2PA	X	X	X	X
WS-6802-2PA With the following PAs: PA-A3-OC3MM PA-A3-T3 PA-MC-T3	X	X		X
WS-SVC-FWM-1	X	X		X
WS-SVC-IDSM2	X	X		
WS-SVC-IDSUPG	X	X		

Table 22-2 Line Card Module Support for the IPSec VPN SPA by Release (continued)

Line Card Module	Cisco IOS Release 12.2(18)SX		Cisco IOS Release 12.2(33)SR	
WS-SVC-NAM2	X	X		
WS-SVC-WEBVPN-K9	X	X		X
WS-X6148-GE-TX	X	X	X	X
WS-X6408A-GBIC	X	X		X
WS-X6416-GBIC	X	X		X
WS-X6416-GE-MT		X		X
WS-X6502-10GE	X	X	X	X
WS-X6516-GBIC	X	X	X	X
WS-X6516-GE-TX	X	X	X	X
WS-X6516A-GBIC	X	X	X	X
WS-X6548-GE-TX	X	X	X	X
WS-X6548V-GE-TX		X		X
WS-X6548-RJ-21		X		X
WS-X6548-RJ-45	X	X	X	X
WS-X6704-10GE	X	X	X	X
WS-X6748-GE-TX	X	X	X	X
WS-X6748-SFP	X	X	X	X

1. Cisco IOS Release 12.2(18)SXF2: Switch port configurations are not supported when a Cisco 7600 SIP-400 is present in the chassis.
2. Cisco IOS Release 12.2(33)SRA: Switch port configurations are not supported when a Cisco 7600 SIP-400 is present in the chassis.
3. Cisco IOS Release 12.2(33)SRA: MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. That is, you must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

Restrictions

The IPSec VPN SPA is subject to the following restrictions:

- The IPSec VPN SPA is only supported on the Cisco 7600 SSC-400.
- The Cisco 7600 SSC-400 is not Route Processor Redundancy Plus (RPR+) or Stateful Switchover (SSO) aware. As a result, the Cisco 7600 SSC-400 will reset if RPR+ or SSO is configured.
- As of Cisco IOS Release 12.2(33)SRA, the IPSec VPN SPA is only supported on a Cisco 7600 series router using a Supervisor Engine 720 (MSFC3 and PFC3) with a minimum of 512 MB memory or a Supervisor Engine 32. For a list of the Supervisor Engine support for each release, see Table 22-1 on page 22-9.
- The IPSec VPN SPA MSFC DRAM requirements are as follows: .
  - Up to 8,000 tunnels with 512-MB DRAM
  - Up to 16,000 tunnels with 1-GB DRAM

These numbers are chosen to leave some memory available for routing protocols and other applications. However, your particular use of the MSFC may demand more memory than the quantities that are listed above. In an extreme case, you could have one tunnel but still require 512-MB DRAM for other protocols and applications running on the MSFC.

- Only the following Cisco 7600 series routers are supported:
  - 7603 router (CISCO7603)
  - 7606 router (CISCO7606)
  - 7609 router (CISCO7609)
  - 7609 router (OSR-7609)
  - 7613 router (CISCO7613)
- Only 10 IPSec VPN SPAs per chassis are supported in crypto-connect mode; only 6 IPSec VPN SPAs per chassis are supported in VRF mode.
- A distinct crypto map is required per interface (you cannot reuse crypto maps).
- As of Cisco IOS Release 12.2(33)SRA, a maximum number of 2000 IPSec tunnels is supported when PKI is configured with the IPSec VPN SPA.

Supported MIBs

The following MIBs are supported as of Cisco IOS Release 12.2(18)SXE2 for the Cisco 7600 SSC-400 and the IPSec VPN SPA on a Cisco 7600 series router:

- CISCO-ENTITY-ASSET-MIB
- CISCO-ENTITY-EXT-MIB
- CISCO-ENTITY-FRU-CONTROL-MIB
- ENTITY-MIB
- OLD-CISCO-CHASSIS-MIB

For more information about MIB support on a Cisco 7600 series router, refer to the *Cisco 7600 Series Internet Router MIB Specifications Guide*, at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/7600mibs/mibguide/index.htm

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

http://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

Transitioning In and Out of Hardware-Based Cryptographic Mode

When you power on a chassis with an IPSec VPN SPA installed or if you insert an IPSec VPN SPA into a chassis, the chassis enters the hardware-based cryptographic mode. In this mode, the encryption and decryption is done by the IPSec VPN SPA. The MSFC stops performing encryption and decryption in software; the software-based cryptographic mode does not function in this mode. In hardware-based cryptographic mode, you must configure an IPSec VPN SPA-specific configuration.

An IPSec VPN SPA in a chassis is active only if power to the SPA is enabled. If the power to the IPSec VPN SPA is disabled, the SPA is in the inactive state.

Use the **hw-module subslot** *slot*/*subslot* **shutdown** [**powered** | **unpowered**] command in global configuration mode to enable or disable power to the IPSec VPN SPA. Note the **powered** option resets power to the specified SPA subslot, and the **unpowered** option disables power to the specified SPA subslot.

Use the [**no**] **power enable module** command to enable or disable power to the Cisco 7600 SSC-400.

When you power on a chassis with an active IPSec VPN SPA, the chassis enters the hardware-based cryptographic mode even if there is no IPSec VPN SPA-specific configuration. If you power on a chassis with an inactive IPSec VPN SPA, the chassis enters the software-based cryptographic mode; any IPSec VPN SPA-specific configurations in the startup configuration are not implemented.

When a chassis is in the hardware-based cryptographic mode, it stays in the hardware-based cryptographic mode even if you remove the IPSec VPN SPA or turn off the power to the IPSec VPN SPA. In these cases, all the packets that normally would have gone through the IPSec VPN SPA are dropped.

If you power on a chassis with no IPSec VPN SPA or an inactive IPSec VPN SPA, the chassis enters the software-based cryptographic mode. Inserting an IPSec VPN SPA into the chassis or enabling power to the IPSec VPN SPA results in a change to the hardware-based cryptographic mode.

To switch from hardware-based cryptographic mode to software-based cryptographic mode, you must remove the IPSec VPN SPA or power off the IPSec VPN SPA and reset the chassis.

After you enter the hardware-based cryptographic mode by inserting one IPSec VPN SPA into the chassis, inserting additional IPSec VPN SPAs has no effect on the cryptographic mode.

Hardware-Based Cryptographic Mode Configuration Guidelines

The configuration guidelines for hardware-based cryptographic mode are as follows:

- IPSec VPN SPA removal

  When you remove an IPSec VPN SPA that has some ports participating in crypto connection, the crypto connections remain intact. When you reinsert the same type of IPSec VPN SPA, the traffic starts to run again on all the crypto connections. You must manually remove the crypto connections that are associated with the removed IPSec VPN SPA. You can enter the **no crypto connect vlan** command from any interface when the associated physical port is removed.

- Rebooting an IPSec VPN SPA with crypto connections

  When you reboot an IPSec VPN SPA that has crypto connections, the existing crypto connections are kept intact. The traffic starts running again when the IPSec VPN SPA reboots. When a crypto connection exists but the associated interface VLAN is missing from the IPSec VPN SPA inside port, the crypto connection is removed after the IPSec VPN SPA reboots.

- When you remove a port VLAN or an interface VLAN with the **no interface vlan** command, the associated crypto connection is also removed.

Displaying the SPA Hardware Type and Statistics

There are several commands on the Cisco 7600 series router that provide IPSec VPN SPA hardware information.

- To verify the SPA hardware type that is installed in your Cisco 7600 series router, use the **show diagbus** command.
- To display hardware information for the IPSec VPN SPA, use the **show crypto eli** command.
- To display platform and network interface controller statistics for the IPSec VPN SPA, use the **show crypto engine accelerator statistic** command.

For more information about these commands, see the *Cisco 7600 Series Cisco IOS Command Reference* publication.

Table 22-3 shows the hardware description that appears in the **show diagbus** command output for an IPSec VPN SPA on the Cisco 7600 series router.

Table 22-3 SPA Hardware Descriptions in show diagbus Command

SPA	Description in show diagbus Command
SPA-IPSEC-2G	"SPA-IPSEC-2G (0x3D7)"

Example of the show diagbus Command

The following example shows output from the **show diagbus** command on a Cisco 7600 series router with an IPSec VPN SPA installed in subslot 1 of a Cisco 7600 SSC-400 that is installed in slot 5:

```
Router# show diagbus

Slot 5: Logical_index 10
        2-subslot Services SPA Carrier-400 controller
        Board is analyzed ipc ready
        HW rev 0.3, board revision A01
        Serial Number: abc Part number: 73-6348-01

        Slot database information:
        Flags: 0x2004   Insertion time: 0x3DB5F4BC (4d20h ago)

        Controller Memory Size:
                248 MBytes CPU Memory
                8 MBytes Packet Memory
                256 MBytes Total on Board SDRAM
        Cisco IOS Software, cwlc Software (smsc-DWDBG-M), Version 12.2(nightly.SRA060615)
NIGHTLY BUILD, synched to rainier RAINER_BASE

        SPA Information:
        subslot 5/1: SPA-IPSEC-2G (0x3D7), status: ok
```

Example of the show crypto eli Command

The following example shows output from the **show crypto eli** command on a Cisco 7600 series router with IPSec VPN SPAs installed in subslots 0 and 1 of a Cisco 7600 SSC-400 that is installed in slot 3. The output displays how many IKE-SAs and IPSec sessions are active and how many Diffie-Hellman keys are in use for each IPSec VPN SPA.

Note The **show crypto eli** command is only supported on the IPSec VPN SPA as of Cisco IOS Release 12.2(33)SRA.

```
Router# show crypto eli

>>Hardware Encryption : ACTIVE
>> Number of hardware crypto engines = 2
>>
>> CryptoEngine SPA-IPSEC-2G[3/0] details: state = Active
>> Capability      :
>>     IPSEC: DES, 3DES, AES, RSA
>>
>> IKE-Session   :     0 active, 16383 max, 0 failed
>> DH            :     0 active,  9999 max, 0 failed
>> IPSec-Session :     0 active, 65534 max, 0 failed
>>
>> CryptoEngine SPA-IPSEC-2G[3/1] details: state = Active
>> Capability      :
>>     IPSEC: DES, 3DES, AES, RSA
>>
>> IKE-Session   :     1 active, 16383 max, 0 failed
>> DH            :     0 active,  9999 max, 0 failed
>> IPSec-Session :     2 active, 65534 max, 0 failed
```

Example of the show crypto engine accelerator statistic Command

The following example shows output from the **show crypto engine accelerator statistic** command on a Cisco 7600 series router with an IPSec VPN SPA in subslot 0 of a Cisco 7600 SSC-400 that is installed in slot 1. The output displays platform statistics for the IPSec VPN SPA and also displays the network interface controller statistics.

Note The **show crypto engine accelerator statistic** command is only supported as of Cisco IOS Release 12.2(33)SRA.

```
Router# show crypto engine accelerator statistic slot 1/0 detail

VPN module in slot 1/0


Decryption Side Data Path Statistics
====================================
Packets RX...............: 454260
Packets TX...............: 452480

IPSec Transport Mode.....: 0
IPSec Tunnel Mode........: 452470
AH Packets...............: 0
```

```
ESP Packets..............: 452470
GRE Decapsulations.......: 0
NAT-T Decapsulations.....: 0
Clear....................: 8
ICMP.....................: 0

Packets Drop.............: 193
Authentication Errors....: 0
Decryption Errors........: 0
Replay Check Failed......: 0
Policy Check Failed......: 0
Illegal CLear Packet.....: 0
GRE Errors...............: 0
SPD Errors...............: 0
HA Standby Drop..........: 0

Hard Life Drop...........: 0
Invalid SA...............: 191
SPI No Match.............: 0
Destination No Match.....: 0
Protocol No Match........: 0

Reassembly Frag RX.......: 0
IPSec Fragments..........: 0
IPSec Reasm Done.........: 0
Clear Fragments..........: 0
Clear Reasm Done.........: 0
Datagrams Drop...........: 0
Fragments Drop...........: 0


Decryption Side Controller Statistics

=====================================
Frames RX................: 756088
Bytes RX.................: 63535848
Mcast/Bcast Frames RX....: 2341
RX Less 128Bytes.........: 756025
RX Less 512Bytes.........: 58
RX Less 1KBytes..........: 2
RX Less 9KBytes..........: 3
RX Frames Drop...........: 0

Frames TX................: 452365
Bytes TX.................: 38001544
Mcast/Bcast Frames TX....: 9
TX Less 128Bytes.........: 452343
TX Less 512Bytes.........: 22
TX Less 1KBytes..........: 0
TX Less 9KBytes..........: 0


Encryption Side Data Path Statistics
====================================
Packets RX...............: 756344
Packets TX...............: 753880
IPSec Transport Mode.....: 0
IPSec Tunnel Mode........: 753869
GRE Encapsulations.......: 0
NAT-T Encapsulations.....: 0
LAF prefragmented........: 0

Fragmented...............: 0
Clear....................: 753904
```

```
ICMP.....................: 0

Packets Drop.............: 123
IKE/TED Drop.............: 27
Authentication Errors....: 0
Encryption Errors........: 0
HA Standby Drop..........: 0

Hard Life Drop...........: 0
Invalid SA...............: 191

Reassembly Frag RX.......: 0
Clear Fragments..........: 0
Clear Reasm Done.........: 0
Datagrams Drop...........: 0
Fragments Drop...........: 0


Encryption Side Controller Statistics
=====================================
Frames RX................: 454065
Bytes RX.................: 6168274/
Mcast/Bcast Frames RX....: 1586
RX Less 128Bytes.........: 1562
RX Less 512Bytes.........: 452503
RX Less 1KBytes..........: 0
RX Less 9KBytes..........: 0
RX Frames Drop...........: 0

Frames TX................: 753558
Bytes TX.................: 100977246
Mcast/Bcast Frames TX....: 2
TX Less 128Bytes.........: 3
TX Less 512Bytes.........: 753555
TX Less 1KBytes..........: 0
TX Less 9KBytes..........: 0
```

CHAPTER 23

Configuring VPNs on the IPSec VPN SPA

This chapter provides information about configuring IPSec VPNs on the IPSec VPN SPA on the Cisco 7600 series router. It includes the following sections:

Note The procedures in this chapter assume you have familiarity with security configuration concepts, such as VLANs, ISAKMP policies, preshared keys, transform sets, access control lists, and crypto maps. For more information about these and other security configuration concepts, refer to the *Cisco IOS Security Configuration Guide and the Cisco IOS Security Command Reference.*

For more information about the installation of cards on the Cisco 7600 series router, refer to the *Cisco 7600 Series Router SIP, SSC, and SPA Hardware Installation Guide* at this URL:
http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/76sipspa/sipspahw/index.htm

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Tip To ensure a successful configuration of your VPN using the IPSec VPN SPA, read all of the configuration summaries and guidelines before you perform any configuration tasks.

Overview of Basic IPSec and IKE Configuration Concepts

This subsection reviews some basic IPSec and IKE concepts that are used throughout the configuration of the IPSec VPN SPA, such as security associations (SAs), access lists (ACLs), crypto maps, transform sets, and IKE policies. The information presented here is introductory and should not be considered complete. For more detailed information on IPSec and IKE concepts and procedures, refer to the *Cisco IOS Security Configuration Guide.*

Information About IPSec Configuration

IPSec provides secure tunnels between two peers, such as two routers. More accurately, these tunnels are sets of security associations (SAs) that are established between two IPSec peers. The SAs define which protocols and algorithms should be applied to sensitive packets and specify the keying material to be used by the two peers. SAs are unidirectional and are established per security protocol (Authentication Header (AH) or Encapsulating Security Payload (ESP)). Multiple IPSec tunnels can exist between two peers to secure different data streams, with each tunnel using a separate set of SAs. For example, some data streams might be authenticated only while other data streams must both be encrypted and authenticated.

Note The use of the term "tunnel" in this subsection does not refer to using IPSec in tunnel mode.

With IPSec, you define what traffic should be protected between two IPSec peers by configuring ACLs and applying these ACLs to interfaces by way of crypto maps. (The ACLs used for IPSec are used only to determine which traffic should be protected by IPSec, not which traffic should be blocked or permitted through the interface. Separate ACLs define blocking and permitting at the interface.)

If you want certain traffic to receive one combination of IPSec protection (for example, authentication only) and other traffic to receive a different combination of IPSec protection (for example, both authentication and encryption), you must create two different crypto ACLs to define the two different types of traffic. These different ACLs are then used in different crypto map entries, which specify different IPSec policies.

Crypto ACLs associated with IPSec crypto map entries have four primary functions:

- Select outbound traffic to be protected by IPSec (permit = protect).
- Indicate the data flow to be protected by the new SAs (specified by a single permit entry) when initiating negotiations for IPSec security associations.
- Process inbound traffic in order to filter out and discard traffic that should have been protected by IPSec.
- Determine whether or not to accept requests for IPSec security associations on behalf of the requested data flows when processing IKE negotiation from the IPSec peer. Negotiation is performed only for ipsec-isakmp crypto map entries. In order to be accepted, if the peer initiates the IPSec negotiation, it must specify a data flow that is "permitted" by a crypto ACL associated with an ipsec-isakmp crypto map entry.

Crypto map entries created for IPSec pull together the various parts used to set up IPSec SAs, including:

- Which traffic should be protected by IPSec (per a crypto ACL)
- The granularity of the flow to be protected by a set of SAs
- Where IPSec-protected traffic should be sent (the name of the remote IPSec peer)

- The local address to be used for the IPSec traffic
- What IPSec SA should be applied to this traffic (selecting from a list of one or more transform sets)
- Whether SAs are manually established or are established via IKE
- Other parameters that might be necessary to define an IPSec SA

Crypto map entries are searched in order—the router attempts to match the packet to the access list specified in that entry.

Crypto map entries also include transform sets. A transform set is an acceptable combination of security protocols, algorithms, and other settings to apply to IPSec-protected traffic.

You can specify multiple transform sets, and then specify one or more of these transform sets in a crypto map entry. During IPSec security association negotiations with IKE, the peers search for a transform set that is the same at both peers. When such a transform set is found, it is selected and will be applied to the protected traffic as part of both peers' IPSec SAs. (With manually established SAs, there is no negotiation with the peer, so both sides must specify the same transform set.)

Information About IKE Configuration

Internet Key Exchange (IKE) is a key management protocol standard that is used in conjunction with the IPSec standard.

IKE is a hybrid protocol that implements the Oakley key exchange and Skeme key exchange inside the Internet Security Association and Key Management Protocol (ISAKMP) framework. (ISAKMP, Oakley, and Skeme are security protocols implemented by IKE.)

IPSec can be configured without IKE, but IKE enhances IPSec by providing additional features, flexibility, and ease of configuration for the IPSec standard. IKE is enabled by default.

You configure IKE by creating IKE policies at each peer using the **crypto isakmp policy** command. An IKE policy defines a combination of security parameters to be used during the IKE negotiation and mandates how the peers are authenticated.

You can create multiple IKE policies, each with a different combination of parameter values, but at least one of these policies must contain exactly the same encryption, hash, authentication, and Diffie-Hellman parameter values as one of the policies on the remote peer. For each policy that you create, you assign a unique priority (1 through 10,000, with 1 being the highest priority).

If you do not configure any policies, your router uses the default policy, which is always set to the lowest priority, and which contains each parameter's default value.

There are five parameters to define in each IKE policy:

- Encryption algorithm
- Hash algorithm
- Authentication method
- Diffie-Hellman group identifier
- Security association lifetime

Configuring VPNs with the IPSec VPN SPA

To configure a VPN using the IPSec VPN SPA, you have two basic options: crypto-connect mode or Virtual Routing and Forwarding (VRF) mode. In either mode, you may also configure GRE tunneling to encapsulate a wide variety of protocol packet types, including multicast packets, inside the VPN tunnel.

Note Switching between crypto-connect mode and VRF mode is not supported.

Crypto-Connect Mode

Traditionally, VPNs are configured on the IPSec VPN SPA by attaching crypto maps to interface VLANs and then crypto-connecting a physical port to the interface VLAN. This method, known as crypto-connect mode, is similar to the method used to configure VPNs on routers running Cisco IOS software. When you configure VPNs on the IPSec VPN SPA using crypto-connect mode, you attach crypto maps to VLANs (using interface VLANs); when you configure VPNs on routers running Cisco IOS software, you configure individual interfaces.

Note With the IPSec VPN SPA, crypto maps are still attached to individual interfaces but the set of interfaces allowed is restricted to interface VLANs.

When configuring a VPN using crypto- connect mode, you have two additional options: GRE tunneling and IP Multicast.

VRF Mode

The VRF-Aware IPSec feature introduces IPSec tunnel mapping to Multiprotocol Label Switching (MPLS) VPNs. Using the VRF-Aware IPSec feature, you can map IPSec tunnels to VPN routing and forwarding instances (VRFs) using a single public-facing address. A VRF instance is a per-VPN routing information repository that defines the VPN membership of a customer site attached to the Provider Edge (PE) router. A VRF comprises an IP routing table, a derived Cisco Express Forwarding (CEF) table, a set of interfaces that use the forwarding table, and a set of rules and routing protocol parameters that control the information that is included in the routing table. A separate set of routing and CEF tables is maintained for each VPN customer.

When you configure a VPN on the IPSec VPN SPA using VRF mode, the model of interface VLANs is preserved, but the **crypto connect vlan** command is not used. Instead, a route must be installed so that packets destined for that particular subnet in that particular VRF are directed to that interface VLAN.

When configuring a VPN using VRF mode, you have these additional tunneling options: tunnel protection (TP) using GRE, and Virtual Tunnel Interface (VTI). When configuring VTI, you can terminate tunnels in VRFs (normal VRF mode) or in the global context, using crypto connect alternative (CCA) mode.

Configuring Ports in Crypto-Connect Mode

Before beginning your crypto-connect mode port configurations, you should read the following subsections:

- Understanding Port Types in Crypto-Connect Mode, page 23-5
- Crypto-Connect Mode Configuration Guidelines and Restrictions, page 23-8

Then perform the procedures in the following subsections:

- Configuring the IPSec VPN SPA Inside Port and Outside Port, page 23-8
- Configuring an Access Port, page 23-9
- Configuring a Routed Port, page 23-13
- Configuring a Trunk Port, page 23-17
- Configuring IPSec VPN SPA Connections to WAN Interfaces, page 23-22
- Displaying the VPN Running State, page 23-23

Note The configuration procedures in this section do not provide GRE tunneling support. For information on how to configure GRE tunneling support in crypto connect mode, see the "Configuring GRE Tunneling in Crypto-Connect Mode" section on page 23-33.

Note The procedures in this section do not provide detailed information on configuring the following Cisco IOS features: IKE policies, preshared key entries, Cisco IOS ACLs, and crypto maps. For detailed information on configuring these features, refer to the following Cisco IOS documentation:

Cisco IOS Security Configuration Guide, *Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/index.htm

Cisco IOS Security Command Reference, *Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_r/index.htm

Understanding Port Types in Crypto-Connect Mode

To configure IPSec VPNs in crypto-connect mode, you should understand the following concepts:

- Router Outside Ports and Inside Ports, page 23-6
- IPSec VPN SPA Outside Port and Inside Port, page 23-6
- Port VLAN and Interface VLAN, page 23-6
- Access Ports, Routed Ports, and Trunk Ports, page 23-7

Router Outside Ports and Inside Ports

The Fast Ethernet or Gigabit Ethernet ports on the Cisco 7600 series router that connect to the WAN routers are referred to as router outside ports. These ports connect the LAN to the Internet or to remote sites. Cryptographic policies are applied to the router outside ports.

The Fast Ethernet or Gigabit Ethernet ports on the Cisco 7600 series router that connect to the LAN are referred to as router inside ports.

The IPSec VPN SPA sends encrypted packets to the router outside ports and decrypted packets to the Policy Feature Card 2 (PFC2) for Layer 3 forwarding to the router inside ports.

IPSec VPN SPA Outside Port and Inside Port

The IPSec VPN SPA appears to the CLI as a SPA with two Gigabit Ethernet ports. The IPSec VPN SPA has no external connectors; the Gigabit Ethernet ports connect the IPSec VPN SPA to the router backplane and Switch Fabric Module (SFM) (if installed).

One Gigabit Ethernet port handles all the traffic going to and coming from the router outside ports. This port is referred to as the *IPSec VPN SPA outside port*. The other Gigabit Ethernet port handles all traffic going to and coming from the LAN or router inside ports. This port is referred to as the *IPSec VPN SPA inside port*.

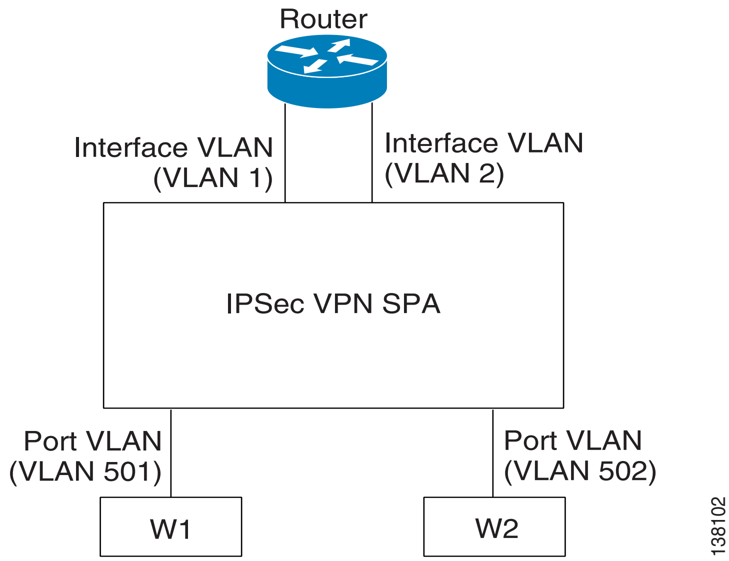
Port VLAN and Interface VLAN

Your VPN configuration can have one or more router outside ports. To handle the packets from multiple router outside ports, you must direct the packets from multiple router outside ports to the IPSec VPN SPA outside port by placing the router outside ports in a VLAN with the outside port of the IPSec VPN SPA. This VLAN is referred to as the *port VLAN*. The port VLAN is a Layer 2-only VLAN. You do not configure Layer 3 addresses or features on this VLAN; the packets within the port VLAN are bridged by the PFC2.

Before the router can forward the packets using the correct routing table entries, the router needs to know which interface that a packet was received on. For each port VLAN, you must create another VLAN so that the packets from every router outside port are presented to the router with the corresponding VLAN ID. This VLAN contains only the IPSec VPN SPA inside port and is referred to as the *interface VLAN*. The interface VLAN is a Layer 3-only VLAN. You configure the Layer 3 address and Layer 3 features, such as ACLs and the crypto map, to the interface VLAN.

After you create and configure the port VLAN and the interface VLAN, you tie the VLANs together using the **crypto connect vlan** command. Figure 23-1 shows an example of the port VLAN and interface VLAN configurations.

Figure 23-1 Port VLAN and Interface VLAN Configuration Example



Port VLAN 501 and port VLAN 502 are the port VLANs that are associated with the router outside ports W1 and W2.

Interface VLAN 1 and interface VLAN 2 are the interface VLANs that correspond to port VLAN 501 and port VLAN 502.

You configure the IP address, ACLs, and crypto map that apply to the router outside port W1 on interface VLAN 1. You configure the features that apply to the router outside port W2 on interface VLAN 2.

Packets coming from the WAN through port W1 (port W1 belongs to port VLAN 501) are directed by the PFC2 to the IPSec VPN SPA outside port. The IPSec VPN SPA decrypts the packets and changes the VLAN to interface VLAN 1 and then presents the packet to the router through the IPSec VPN SPA inside port. The PFC2 then routes the packet to the proper destination.

Packets going from the LAN to the outside ports are first routed by the PFC2. Based on the route, the PFC2 routes the packets to one of the interface VLANs and directs the packet to the IPSec VPN SPA inside port. The IPSec VPN SPA applies the cryptographic policies that are configured on the corresponding interface VLAN, encrypts the packet, changes the VLAN ID to the corresponding port VLAN, and sends the packet to the router outside port through the IPSec VPN SPA outside port.

Access Ports, Routed Ports, and Trunk Ports

The IPSec VPN SPA supports three types of Ethernet connections:

- Access ports—Access ports have an external or VLAN Trunk Protocol (VTP) VLAN associated with them. You can associate more than one port to a defined VLAN.
- Routed ports—By default, all Ethernet ports are routed ports. These ports have a hidden VLAN associated with them.
- Trunk ports—Trunk ports carry many external or VTP VLANs, on which all packets are encapsulated with an 802.1Q header.

Crypto-Connect Mode Configuration Guidelines and Restrictions

Follow these guidelines and restrictions to prevent IPSec VPN SPA misconfigurations when configuring VPN ports in crypto-connect mode:

- Switch port configurations are not supported when a Cisco 7600 SIP-400 is present in the chassis.
- Be careful about removing a line in a crypto ACL because removing a line causes all crypto maps using that ACL to be removed and reattached to the IPSec VPN SPA. This action causes intermittent connectivity problems for all the security associations (SAs) derived from the crypto maps that reference that ACL.
- Do not attach a crypto map set to a loopback interface. However, you can maintain an IPSec security association database independent of physical ingress and egress interfaces with the IPSec VPN SPA by entering the **crypto map local-address** command.

  If you apply the same crypto map set to each secure interface and enter the **crypto map local-address** command with the interface as a loopback interface, you will have a single security association database for the set of secure interfaces.
- Be aware that if you configure a crypto map with an empty ACL (an ACL that is defined but has no lines) and attach the crypto map to an interface, all traffic going out of that interface is dropped. However, with the IPSec VPN SPA, all traffic goes out of the interface in the clear (unencrypted) state.
- Do not convert existing crypto-connected port characteristics. When the characteristics of a crypto-connected access port or a routed port change (switch port to routed port or vice versa), the associated crypto connection is deleted.
- Do not remove the interface VLAN or port VLAN from the VLAN database. All interface VLANs and port VLANs must be in the VLAN database. When you remove these VLANs from the VLAN database, the running traffic stops.

  When you enter the **crypto connect vlan** command and the interface VLAN or port VLAN is not in the VLAN database, the following warning message is displayed:

  ```
  VLAN id 100 not found in current VLAN database. It may not function correctly unless
  VLAN 100 is added to VLAN database.
  ```
- When replacing a crypto map on an interface, always enter the **no crypto map** command before reapplying a crypto map on the interface.
- Be aware that after a Supervisor Engine switchover, the installed SPAs reboot and come back online. During this period, the IPSec VPN SPA's established security associations (SAs) are temporarily lost and are reconstructed after the SPA comes back online. The reconstruction is through IKE (it is not instantaneous).
- Do not use IPSec stateful failover using a Blade Failure Group (BFG) in crypto-connect mode with tunnel protection.

Configuring the IPSec VPN SPA Inside Port and Outside Port

In most cases, you do not explicitly configure the IPSec VPN SPA inside and outside ports. Cisco IOS software configures these ports automatically.

IPSec VPN SPA Inside and Outside Port Configuration Guidelines and Restrictions

Follow these guidelines for configuring the IPSec VPN SPA inside and outside ports:

- Do not configure the IPSec VPN SPA outside port. Cisco IOS software configures the port automatically.
- Do not configure the inside trunk port. Cisco IOS software configures the port automatically based on the **crypto engine slot** command.
- Do not change the port characteristics of the IPSec VPN SPA inside port unless it is necessary to set the trusted state.

Note Although the default trust state of the inside port is trusted, certain global settings may cause the state to change. You may need to configure the **mls qos trust** command on the inside port to set the interface to the trusted state.

If you accidentally change the inside port characteristics, enter the following commands to return the port characteristics to the defaults:

```
Router(config-if)# switchport
Router(config-if)# no switchport access vlan
Router(config-if)# switchport trunk allowed vlan 1,1002-1005
Router(config-if)# switchport trunk encapsulation dot1q
Router(config-if)# switchport mode trunk
```

- Do not remove a VLAN from the IPSec VPN SPA inside port. The running traffic stops when you remove an interface VLAN from the IPSec VPN SPA inside port while the crypto connection to the interface VLAN exists. The crypto connection is not removed and the **crypto connect vlan** command still shows up in the **show running-config** command display. If you enter the **write memory** command with this running configuration, your startup-configuration file would be misconfigured.

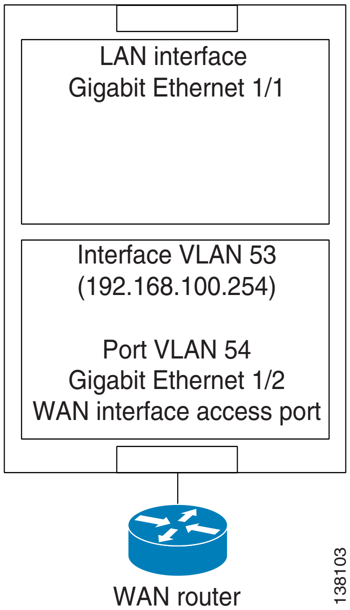
Note It is not possible to remove an interface VLAN from the IPSec VPN SPA inside port while the crypto connection to the interface VLAN exists. You must first remove the crypto connection.

- Do not remove a VLAN from the IPSec VPN SPA outside port. The running traffic stops when you remove a port VLAN from the IPSec VPN SPA outside port while the crypto connection to the interface VLAN exists. The crypto connection is not removed and the **crypto connect vlan** command still shows up in the **show running-config** command display. Removing a VLAN from the IPSec VPN SPA outside port does not affect anything in the startup-configuration file because the port VLAN is automatically added to the IPSec VPN SPA outside port when the **crypto connect vlan** command is entered.

Configuring an Access Port

This section describes how to configure the IPSec VPN SPA with an access port connection to the WAN router (see Figure 23-2).

Figure 23-2 Access Port Configuration Example



Note Switch port connections are not supported when a Cisco 7600 SIP-400 is present in the chassis.

To configure an access port connection to the WAN router, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp policy** *priority* **...** Router(config-isakmp) # **exit**	Defines an ISAKMP policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest. For details on configuring an ISAKMP policy, see the *Cisco IOS Security Configuration Guide*.
Step 2	Router(config)# **crypto isakmp key** *keystring* **address** *peer-address*	Configures a preshared authentication key. • *keystring*—Preshared key. • *peer-address*—IP address of the remote peer. For details on configuring a preshared key, see the *Cisco IOS Security Configuration Guide*.

	Command	Purpose
Step 3	Router(config)# **crypto ipsec transform-set** *transform-set-name transform1*[*transform2*[*transform3*]] **...** Router(config-crypto-tran)# **exit**	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPSec security protocols and algorithms. For accepted *transformx* values, and more details on configuring transform sets, see the *Cisco IOS Security Command Reference*.
Step 4	Router(config)# **access list** *access-list-number* {**deny** | **permit**} **ip** *source source-wildcard destination destination-wildcard*	Defines an extended IP access list. • *access-list-number*—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699. • {**deny** | **permit**}—Denies or permits access if the conditions are met. • *source*—Address of the host from which the packet is being sent. • *source-wildcard*—Wildcard bits to be applied to the source address. • *destination*—Address of the host to which the packet is being sent. • *destination-wildcard*—Wildcard bits to be applied to the destination address. For details on configuring an access list, see the *Cisco IOS Security Configuration Guide*.
Step 5	Router(config)# **crypto map** *map-name seq-number* **ipsec-isakmp** **...** Router(config-crypto-map)# **exit**	Creates or modifies a crypto map entry and enters the crypto map configuration mode. • *map-name*—Name that identifies the crypto map set. • *seq-number*—Sequence number you assign to the crypto map entry. Lower values have higher priority. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec security associations. For details on configuring a crypto map, see the *Cisco IOS Security Configuration Guide*.
Step 6	Router(config)# **vlan** *inside-vlan-id*	Adds the VLAN ID into the VLAN database. • *inside-vlan-id*—VLAN identifier.
Step 7	Router(config)# **vlan** *outside-vlan-id*	Adds the VLAN ID into the VLAN database. • *outside-vlan-id*—VLAN identifier.

	Command	Purpose
Step 8	Router(config)# **interface vlan** *inside-vlan-id*	Enters interface configuration mode for the specified VLAN interface. • *inside-vlan-id*—VLAN identifier.
Step 9	Router(config-if)# **description inside_interface_vlan_for_crypto_map**	(Optional) Adds a comment to help identify the interface.
Step 10	Router(config-if)# **ip address** *address mask*	Specifies the IP address and subnet mask for the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 11	Router(config-if)# **crypto map** *map-name*	Applies a previously defined crypto map set to the interface. • *map-name*—Name that identifies the crypto map set. Enter the *map-name* value you created in Step 5.
Step 12	Router(config-if)# **no shutdown**	Enables the interface as a Layer 3 inside interface VLAN.
Step 13	Router(config-if)# **crypto engine slot** *slot/subslot*	Assigns the crypto engine to the inside interface VLAN. • *slot*/*subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 14	Router(config)# **interface vlan** *outside-vlan-id*	Enters interface configuration mode for the specified VLAN interface. • *outside-vlan-id*—VLAN identifier.
Step 15	Router(config-if)# **description outside_access_vlan**	(Optional) Adds a comment to help identify the interface.
Step 16	Router(config-if)# **no shutdown**	Enables the interface as an outside access port VLAN.
Step 17	Router(config-if)# **crypto connect vlan** *inside-vlan-id*	Connects the outside access port VLAN to the inside interface VLAN and enters crypto-connect mode. • *inside-vlan-id*—VLAN identifier.
Step 18	Router(config-if)# **interface gigabitethernet** *slot*/*subslot*/*port*	Enters interface configuration mode for the secure port. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 19	Router(config-if)# **description outside_secure_port**	(Optional) Adds a comment to help identify the interface.
Step 20	Router(config-if)# **switchport**	Configures the interface for Layer 2 switching.

	Command	Purpose
Step 21	Router(config-if)# **switchport access vlan** *outside-vlan-id*	Specifies the default VLAN for the interface. • *outside-vlan-id*—VLAN identifier.
Step 22	Router(config-if)# **exit**	Exits interface configuration mode.

For access port configuration examples, see the "Access Port (Crypto-Connect Mode) Configuration Examples" section on page 23-50.

Verifying the Access Port Configuration

To verify an access port configuration, enter the **show crypto vlan** command.

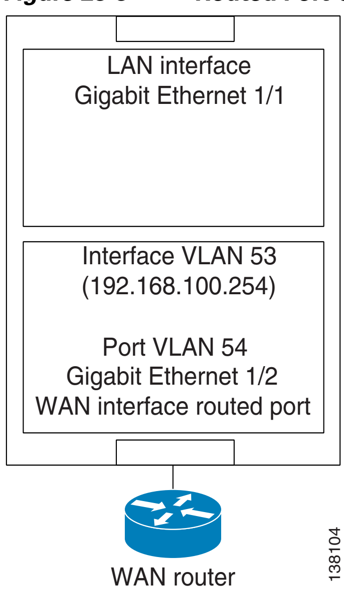
```
Router# show crypto vlan

Interface VLAN 100 on IPSec Service Module port Gi5/0/1 connected to VLAN 2022 with crypto
map set coral2
```

Configuring a Routed Port

This section describes how to configure the IPSec VPN SPA with a routed port connection to the WAN router (see Figure 23-3).

Note A routed port uses a hidden VLAN.

Figure 23-3 Routed Port Configuration Example



Note Switch port connections are not supported when a Cisco 7600 SIP-400 is present in the chassis.

Routed Port Configuration Guidelines

Follow these configuration guidelines for configuring a routed port using the IPSec VPN SPA:

- When a routed port has a crypto connection, the IP ACLs that are attached to the routed port work correctly even if the routed port does not have an IP address.

To configure a routed port connection to the WAN router, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp policy** *priority* **...** Router(config-isakmp) # **exit**	Defines an ISAKMP policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest. For details on configuring an ISAKMP policy, see the *Cisco IOS Security Configuration Guide*.
Step 2	Router(config)# **crypto isakmp key** *keystring* **address** *peer-address*	Configures a preshared authentication key. • *keystring*—Preshared key. • *peer-address*—IP address of the remote peer. For details on configuring a preshared key, see the *Cisco IOS Security Configuration Guide*.
Step 3	Router(config)# **crypto ipsec transform-set** *transform-set-name transform1*[*transform2*[*transform3*]] **...** Router(config-crypto-tran)# **exit**	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPSec security protocols and algorithms. For accepted *transformx* values, and more details on configuring transform sets, see the *Cisco IOS Security Command Reference*.

<table>
<tr><th></th><th>Command</th><th>Purpose</th></tr>
<tr><td>Step 4</td><td>Router(config)# access list access-list-number {deny | permit} ip source source-wildcard destination destination-wildcard</td><td>Defines an extended IP access list.
• access-list-number—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699.
• {deny | permit}—Denies or permits access if the conditions are met.
• source—Address of the host from which the packet is being sent.
• source-wildcard—Wildcard bits to be applied to the source address.
• destination—address of the host to which the packet is being sent.
• destination-wildcard—Wildcard bits to be applied to the destination address.
For details on configuring an access list, see the Cisco IOS Security Configuration Guide.</td></tr>
<tr><td>Step 5</td><td>Router(config)# crypto map map-name seq-number ipsec-isakmp
...
Router(config-crypto-map)# exit</td><td>Creates or modifies a crypto map entry and enters the crypto map configuration mode.
• map-name—Name that identifies the crypto map set.
• seq-number—Sequence number you assign to the crypto map entry. Lower values have higher priority.
• ipsec-isakmp— Indicates that IKE will be used to establish the IPSec security associations.
For details on configuring a crypto map, see the Cisco IOS Security Configuration Guide.</td></tr>
<tr><td>Step 6</td><td>Router(config)# vlan inside-vlan-id</td><td>Adds the VLAN ID into the VLAN database.
• inside-vlan-id—VLAN identifier.</td></tr>
<tr><td>Step 7</td><td>Router(config)# interface vlan inside-vlan-id</td><td>Enters interface configuration mode for the specified VLAN interface.
• inside-vlan-id—VLAN identifier.</td></tr>
<tr><td>Step 8</td><td>Router(config-if)# description inside_interface_vlan_for_crypto_map</td><td>(Optional) Adds a comment to help identify the interface.</td></tr>
<tr><td>Step 9</td><td>Router(config-if)# ip address address mask</td><td>Specifies the IP address and subnet mask for the interface.
• address—IP address.
• mask—Subnet mask.</td></tr>
</table>

	Command	Purpose
Step 10	Router(config-if)# **crypto map** *map-name*	Applies a previously defined crypto map set to the interface. • *map-name*—Name that identifies the crypto map set. Enter the *map-name* value you created in Step 5.
Step 11	Router(config-if)# **no shutdown**	Enables the interface as a Layer 3 inside interface VLAN.
Step 12	Router(config-if)# **crypto engine slot** *slot/subslot*	Assigns the crypto engine to the inside interface VLAN. • *slot*/*subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 13	Router(config-if)# **interface gigabitethernet** *slot*/*subslot*/*port*	Enters interface configuration mode for the secure port. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 14	Router(config-if)# **description outside_secure_port**	(Optional) Adds a comment to help identify the interface.
Step 15	Router(config-if)# **crypto connect vlan** *inside-vlan-id*	Connects the routed port to the inside interface VLAN and enters crypto-connect mode. • *inside-vlan-id*—VLAN identifier.
Step 16	Router(config-if)# **exit**	Exits interface configuration mode.

For routed port configuration examples, see the "Routed Port (Crypto-Connect Mode) Configuration Examples" section on page 23-54.

Verifying a Routed Port Configuration

To verify a route port configuration, enter the **show crypto vlan** command. In the following example, the port Gi 2/8 is the crypto-connected port.

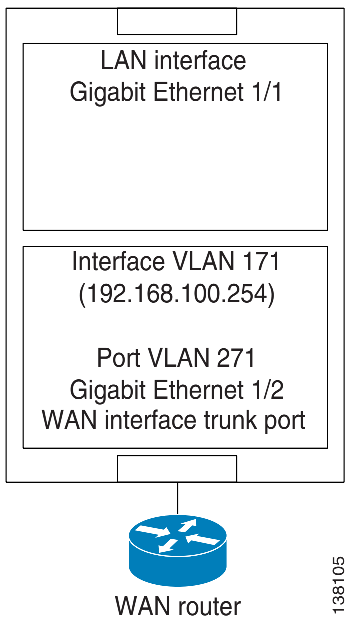
```
Router# show crypto vlan

Interface VLAN 100 on IPSec Service Module port Gi5/0/1 connected to Gi2/8 with crypto
mark set M10K
```

Configuring a Trunk Port

Caution When you configure an Ethernet port as a trunk port, all the VLANs are allowed on the trunk port by default. This default configuration does not work well with the IPSec VPN SPA and causes network loops.

This section describes how to configure the IPSec VPN SPA with a trunk port connection to the WAN router (see Figure 23-4).

Figure 23-4 Trunk Port Configuration Example



Note Switch port connections are not supported when a Cisco 7600 SIP-400 is present in the chassis.

Trunk Port Configuration Guidelines

Follow these configuration guidelines for configuring a trunk port using the IPSec VPN SPA:

- When you configure a trunk port for cryptographic connection, do not use the “all VLANs allowed” default. You need to explicitly specify all the desirable VLANs using the **switchport trunk allowed vlan** command.
- Due to an incorrect startup configuration or through the default trunk port configuration, an interface VLAN might be associated with a trunk port. When you try to remove the interface VLAN from the VLAN list, you might receive an error message similar to the following:

```
Command rejected:VLAN 61 is crypto connected to V162.
```

To remove the interface VLAN from the VLAN list, enter the following commands:

```
Router# configure terminal
Router(config)# interface gigabitethernet1/1
Router(config-if)# no switchport mode trunk
Router(config-if)# switchport trunk allowed vlan 1
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan 1,53,1002-1005
```

Note VLANs in the VLAN list must not include any interface VLANs.

- To avoid getting into the above situation, when you put an Ethernet port into the trunk mode, enter the following commands in the exact order given:

```
Router# configure terminal
Router(config)# interface gigabitethernet1/1
Router(config)# no shut
Router(config-if)# switchport
Router(config-if)# switchport trunk allowed vlan 1
Router(config-if)# switchport trunk encapsulation dot1q
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan 1,53,1002-1005
```

Note VLANs in the VLAN list must not include any interface VLANs.

- A common mistake when configuring a trunk port occurs when you use the **add** option as follows:

```
Router(config-if)# switchport trunk allowed vlan add 100
```

If the **switchport trunk allowed vlan** command has not already been used, the **add** option does not make VLAN 100 the only allowed VLAN on the trunk port; all VLANs are still allowed after entering the command because all the VLANs are allowed by default. After you use the **switchport trunk allowed vlan** command to add a VLAN, you can then use the **switchport trunk allowed vlan add** command to add additional VLANs.

- To remove unwanted VLANs from a trunk port, use the **switchport trunk allowed vlan remove** command.

Caution Do not enter the **switchport trunk allowed vlan all** command on a secured trunk port. In addition, do not set the IPSec VPN SPA inside and outside ports to "all VLANs allowed."

To configure a trunk port connection to the WAN router, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp policy** *priority* **...** Router(config-isakmp) # **exit**	Defines an ISAKMP policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest. For details on configuring an ISAKMP policy, see the *Cisco IOS Security Configuration Guide*.
Step 2	Router(config)# **crypto isakmp key** *keystring* **address** *peer-address*	Configures a preshared authentication key. • *keystring*—Preshared key. • *peer-address*—IP address of the remote peer. For details on configuring a preshared key, see the *Cisco IOS Security Configuration Guide*.
Step 3	Router(config)# **crypto ipsec transform-set** *transform-set-name transform1*[*transform2*[*transform3*]] **...** Router(config-crypto-tran)# **exit**	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPSec security protocols and algorithms. For accepted *transformx* values, and more details on configuring transform sets, see the *Cisco IOS Security Command Reference*.
Step 4	Router(config)# **access list** *access-list-number* {**deny** | **permit**} **ip** *source source-wildcard destination destination-wildcard*	Defines an extended IP access list. • *access-list-number*—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699. • {**deny** | **permit**}—Denies or permits access if the conditions are met. • *source*—Address of the host from which the packet is being sent. • *source-wildcard*—Wildcard bits to be applied to the source address. • *destination*—Address of the host to which the packet is being sent. • *destination-wildcard*—Wildcard bits to be applied to the destination address. For details on configuring an access list, see the *Cisco IOS Security Configuration Guide*.

	Command	Purpose
Step 5	Router(config)# **crypto map** *map-name seq-number* **ipsec-isakmp** **...** Router(config-crypto-map)# **exit**	Creates or modifies a crypto map entry and enters the crypto map configuration mode. • *map-name*—Name that identifies the crypto map set. • *seq-number*—Sequence number you assign to the crypto map entry. Lower values have higher priority. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec security associations. For details on configuring a crypto map, see the *Cisco IOS Security Configuration Guide*.
Step 6	Router(config)# **vlan** *inside-vlan-id*	Adds the VLAN ID into the VLAN database. • *inside-vlan-id*—VLAN identifier.
Step 7	Router(config)# **vlan** *outside-vlan-id*	Adds the VLAN ID into the VLAN database. • *outside-vlan-id*—VLAN identifier.
Step 8	Router(config)# **interface vlan** *inside-vlan-id*	Enters interface configuration mode for the specified VLAN interface. • *inside-vlan-id*—VLAN identifier.
Step 9	Router(config-if)# **description inside_interface_vlan_for_crypto_map**	(Optional) Adds a comment to help identify the interface.
Step 10	Router(config-if)# **ip address** *address mask*	Specifies the IP address and subnet mask for the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 11	Router(config-if)# **crypto map** *map-name*	Applies a previously defined crypto map set to the interface. • *map-name*—Name that identifies the crypto map set. Enter the *map-name* value you created in Step 5.
Step 12	Router(config-if)# **no shutdown**	Enables the interface as a Layer 3 inside interface VLAN.
Step 13	Router(config-if)# **crypto engine slot** *slot/subslot*	Assigns the crypto engine to the inside interface VLAN. • *slot/subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 14	Router(config)# **interface vlan** *outside-vlan-id*	Adds the specified VLAN interface as an outside trunk port VLAN and enters interface configuration mode for the specified VLAN interface. • *outside-vlan-id*—VLAN identifier.
Step 15	Router(config-if)# **description outside_trunk_port_vlan**	(Optional) Adds a comment to help identify the interface.

	Command	Purpose
Step 16	Router(config-if)# **crypto connect vlan** *inside-vlan-id*	Connects the outside trunk port VLAN to the inside interface VLAN and enters crypto-connect mode. • *inside-vlan-id*—VLAN identifier.
Step 17	Router(config-if)# **no shutdown**	Enables the interface as a Layer 3 inside interface VLAN.
Step 18	Router(config-if)# **interface gigabitethernet** *slot/subslot/port*	Enters interface configuration mode for the secure port. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 19	Router(config-if)# **description outside_secure_port**	(Optional) Adds a comment to help identify the interface.
Step 20	Router(config-if)# **switchport**	Configures the interface for Layer 2 switching.
Step 21	Router(config-if)# **no switchport access vlan**	Resets the access VLAN to the appropriate default VLAN for the device.
Step 22	Router(config-if)# **switchport trunk encapsulation dot1q**	Sets the trunk encapsulation to 802.1Q.
Step 23	Router(config-if)# **switchport mode trunk**	Specifies a trunk VLAN Layer 2 interface.
Step 24	Router(config-if)# **switchport trunk allowed vlan remove** *vlan-list*	Removes the specified list of VLANs from those currently set to transmit from this interface. • *vlan-list*—List of VLANs that transmit the interface in tagged format when in trunking mode. Valid values are from 1 to 4094.
Step 25	Router(config-if)# **switchport trunk allowed vlan add** *vlan-list*	Adds the specified VLAN to the list of VLANs currently set to transmit from this interface. • *vlan-list*—List of VLANs that transmit the interface in tagged format when in trunking mode. Valid values are from 1 to 4094.
Step 26	Router(config-if)# **exit**	Exits interface configuration mode.

For trunk port configuration examples, see the "Trunk Port (Crypto-Connect Mode) Configuration Examples" section on page 23-58.

Verifying the Trunk Port Configuration

To verify the VLANs allowed by a trunk port, enter the **show interfaces trunk** command. The following display shows that all VLANs are allowed:

```
Router# show interfaces GigabitEthernet 2/1/1 trunk

Port      Mode         Encapsulation  Status        Native vlan
Gi2/1     on           802.1q         trunking      1

Port      Vlans allowed on trunk
```

```
Gi2/1     1-4094

Port      Vlans allowed and active in management domain
Gi2/1     1-4,7-8,513,1002-1005

Port      Vlans in spanning tree forwarding state and not pruned
Gi2/1     1-4,7-8,513,1002-1005
```

Configuring IPSec VPN SPA Connections to WAN Interfaces

The configuration of IPSec VPN SPA connections to WAN interfaces is similar to the configuration of Ethernet routed interfaces.

IPSec VPN SPA Connections to WAN Interfaces Configuration Guidelines and Restrictions

Follow these guidelines and note these restrictions when configuring a connection to a WAN interface using an IPSec VPN SPA:

- To configure an IPSec VPN SPA connection to a WAN interface, make a crypto connection from the WAN subinterface to the interface VLAN as follows:

```
Router(config)# interface Vlan101
Router(config-if)# ip address 192.168.101.1 255.255.255.0
Router(config-if)# no mop enabled
Router(config-if)# crypto map cwan

Router(config)# interface ATM6/0/0.101 point-to-point
Router(config-subif)# pvc 0/101
Router(config-subif)# crypto connect vlan 101
```

- You must configure a crypto connection on subinterfaces for ATM and Frame Relay.
- For ATM, there is no SVC support, no RFC 1483 bridging, and no point-to-multipoint support.
- For Frame Relay, there is no SVC support, no RFC 1490 bridging, and no point-to-multipoint support.
- For Point-to-Point Protocol (PPP) and Multilink PPP (MLP), you must make the physical interface passive for routing protocols, as follows:

```
Router(config)# router ospf 10
Router(config)# passive-interface multilink1
```

- For PPP and MLP, an **ip unnumbered Null0** command is automatically added to the port configuration to support IPCP negotiation. If you configure a **no ip address** command on the WAN port in the startup configuration, the **no ip address** command will be automatically removed in the running configuration so that it does not conflict with the automatic configuration.
- For PPP and MLP, there is no Bridging Control Protocol (BCP) support.
- When enabled on an inside VLAN, OSPF will be configured in broadcast network mode by default, even when a point-to-point interface (such as T1, POS, serial, or ATM) is crypto-connected to the inside VLAN. However, if OSPF is configured in point-to-point network mode on the peer router (for example, a transit router with no crypto card), OSPF will not establish full adjacency. In this case, you can manually configure OSPF network point-to-point mode in the inside VLAN:

```
Router(config)# interface vlan inside-vlan
Router(config-if)# ip ospf network point-to-point
```

For IPSec VPN SPA connections to WAN interfaces configuration examples, see the "IPSec VPN SPA Connections to WAN Interfaces (Crypto-Connect Mode) Configuration Examples" section on page 23-62.

Displaying the VPN Running State

After you have completed your crypto-connect mode configurations, you can use the **show crypto vlan** command to display the VPN running state. The following examples show the **show crypto vlan** command output for a variety of IPSec VPN SPA configurations.

In the following example, the interface VLAN belongs to the IPSec VPN SPA inside port:

```
Router# show crypto vlan

Interface VLAN 2 on IPSec Service Module port Gi7/1/1 connected to Fa8/3
```

In the following example, VLAN 2 is the interface VLAN and VLAN 2022 is the hidden VLAN:

```
Router# show crypto vlan

Interface VLAN 2 on IPSec Service Module port Gi3/1/1 connected to VLAN 2022 with crypto
map set coral2
```

In the following example, the interface VLAN is missing on the IPSec VPN SPA inside port, the IPSec VPN SPA is removed from the chassis, or the IPSec VPN SPA was moved to a different subslot:

```
Router# show crypto vlan

Interface VLAN 2 connected to VLAN 3 (no IPSec Service Module attached)
```

Configuring VPNs in VRF Mode

The VRF-Aware IPSec feature, known as VRF mode, allows you to map IPSec tunnels to VPN routing and forwarding instances (VRFs) using a single public-facing address.

A VRF instance is a per-VPN routing information repository that defines the VPN membership of a customer site attached to the Provider Edge (PE) router. A VRF comprises an IP routing table, a derived Cisco Express Forwarding (CEF) table, a set of interfaces that use the forwarding table, and a set of rules and routing protocol parameters that control the information that is included in the routing table. A separate set of routing and CEF tables is maintained for each VPN customer.

The MPLS distribution protocol is a high-performance packet-forwarding technology that integrates the performance and traffic management capabilities of data link layer switching with the scalability, flexibility, and performance of network-layer routing.

Note Front door VRF (FVRF) is only supported as of Cisco IOS Release 12.2(33)SRA and later.

Each IPSec tunnel is associated with two VRF domains. The outer encapsulated packet belongs to one VRF domain, called the Front Door VRF (FVRF), while the inner, protected IP packet belongs to another domain called the Inside VRF (IVRF). Another way of stating the same thing is that the local endpoint of the IPSec tunnel belongs to the FVRF while the source and destination addresses of the inside packet belong to the IVRF.

One or more IPSec tunnels can terminate on a single interface. The FVRF of all these tunnels is the same and is set to the VRF that is configured on that interface. The IVRF of these tunnels can be different and depends on the VRF that is defined in the Internet Security Association and Key Management Protocol (ISAKMP) profile that is attached to a crypto map entry.

With the VRF-Aware IPSec feature, packets belonging to a specific VRF are routed through the IPSec VPN SPA for IPSec processing. Through the CLI, you associate a VRF with an interface VLAN that has been configured to point to the IPSec VPN SPA. An interface VLAN must be created for each VRF. Packets traveling from an MPLS cloud to the Internet that are received from an inside VRF are routed to an interface VLAN and then to the IPSec VPN SPA for IPSec processing. The IPSec VPN SPA modifies the packets so that they are placed on a special Layer 3 VLAN for routing to the WAN-side port after they leave the IPSec VPN SPA.

Note Inside VRFs are the VRFs on the unprotected (LAN) side.

Packets traveling in the inbound direction from a protected port on which the **crypto engine slot outside** command has been entered hit a a special ACL and are redirected to the IPSec VPN SPA where they are processed according to the Security Parameter Index (SPI) contained in the packet's IPSec header. Processing on the IPSec VPN SPA ensures that the decapsulated packet is mapped to the appropriate interface VLAN corresponding to the inside VRF. This interface VLAN has been associated with a specific VRF, so packets are routed within the VRF to the correct inside interface.

Note Tunnel protection is supported in VRF mode. For information on configuring tunnel protection, see the "Configuring VPNs in VRF Mode with Tunnel Protection" section on page 23-35 and the "VRF Mode with Tunnel Protection Configuration Example" section on page 23-85.

The following subsections describe how to configure a VPN in VRF mode without tunnel protection on the IPSec VPN SPA.

Understanding VPN Configuration in VRF Mode

In the traditional crypto-connect mode, a VPN is configured by attaching crypto maps to interface VLANs and then crypto-connecting a physical port to the interface VLAN. When configuring a VPN in VRF mode using the IPSec VPN SPA, the idea of interface VLANs is preserved, but the **crypto connect vlan** CLI command is not used. When a packet comes into an interface on a specific VRF, the packet must get to the proper interface VLAN. A route must be installed so that packets destined for that particular subnet in that particular VRF are directed to that interface VLAN. This can be achieved through the following configuration options:

- Configuring an IP address on the interface VLAN that is in the same subnet as the packets' destination IP address. For example, packets are trying to reach subnet 10.1.1.x and their destination IP address is 10.1.1.1 as follows:

```
int vlan 100
 ip vrf forwarding coke
 ip address 10.1.1.254  255.255.255.0 <-- same subnet as 10.1.1.x that we are trying
to reach.
 crypto map mymap
 crypto engine slot 4/0 inside
```

- Configuring a static route as follows:

```
ip route vrf coke 10.1.1.0 255.255.255.0 vlan 100
```

- Configuring routing protocols. You configure the Border Gateway Protocol (BGP), Open Shortest Path First (OSPF), or other routing protocols so that remote routers broadcast their routes.

Note Do not configure routing protocols unless you are using tunnel protection.

- Configuring Reverse Route Injection (RRI). You configure RRI so that a route gets installed when the remote end initiates an IPSec session (as in remote access situations).

With VRF mode, the router sees the interface VLAN as a point-to-point connection; the packets are placed directly onto the interface VLAN. Each VRF has its own interface VLAN.

When a crypto map is attached to an interface VLAN and the **ip vrf forwarding** command has associated that VLAN with a particular VRF, the software creates a point-to-point connection so that all routes pointing to the interface VLAN do not attempt to run the Address Resolution Protocol (ARP). Through normal routing within the VRF, packets to be processed by the IPSec VPN SPA are sent to the interface VLAN. You may configure features on the interface VLAN. The IP address of the interface VLAN must be on the same subnet as the desired destination subnet for packets to be properly routed.

Entering the **ip vrf forwarding** command on an inside interface ensures that all packets coming in on that interface are routed correctly within that VRF.

When you enter the **crypto engine slot outside** command on an interface and enable the **crypto engine mode vrf** command, a special ACL is installed that forces all incoming Encapsulating Security Payload (ESP)/Authentication Header (AH) IPSec packets addressed to a system IP address to be sent to the IPSec VPN SPA WAN-side port. NAT Traversal (NAT-T) packets are also directed to the IPSec VPN SPA by the special ACL.

Note You must enter the **vrf** command from within the context of an ISAKMP profile. This command does not apply to the VRF-aware crypto infrastructure; it applies only to generic crypto processing. When the ISAKMP profile is added to a crypto map set, the VRF becomes the default VRF for all of the crypto maps in the list. Individual crypto maps may override this default VRF by specifying another policy profile that contains a different VRF. If no profile is applied to a crypto map tag, it inherits the VRF from the interface if you have configured the interface with the **ip vrf forwarding** command.

All packets destined for a protected outside interface received in this VRF context are placed on the associated interface VLAN. Similarly, all decapsulated ingress packets associated with this VRF are placed on the appropriate interface VLAN so that they may be routed in the proper VRF context.

VRF Mode Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring a VPN for the IPSec VPN SPA using VRF mode:

- As of Cisco IOS Release 12.2(33)SRA, the **crypto engine subslot** command used in previous releases has been replaced with the **crypto engine slot** command (of the form **crypto engine slo**t *slot*/*sublot* {**inside** | **outside**}). The **crypto engine subslot** command is no longer supported. When upgrading, ensure that this command has been modified in your start-up configuration to avoid extended maintenance time.
- As of Cisco IOS Release 12.2(33)SRA, the **ip vrf forwarding** command is no longer required when configuring GRE with tunnel protection.

- MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF forwarding is to be enabled. That is, you must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.
- When you create an ISAKMP profile, note the following guidelines regarding the use of the **vrf** command:
  - You must use the **vrf** command if you are using the ISAKMP profile with a crypto map.
  - You do not have to use the **vrf** command if you are using the ISAKMP profile with tunnel protection.
  - You should not use the **vrf** command if you are using the ISAKMP profile with DMVPN.

Note After enabling or disabling VRF mode using the **crypto engine mode vrf** command, you must reload the supervisor engine.

- Supported features in VRF mode are as follows:
  - Remote access into a VRF (provider edge [PE]) with the following:
    - Reverse Route Injection (RRI)
    - Proxy AAA (one VRF is proxied to a dedicated AAA)
  - Customer edge-provider edge (CE-PE) encryption using tunnel protection with the following:
    - Routing update propagation between CEs
    - IGP/eBGP routing update propagation between the PE and CEs
  - Overlapping IP address space in VRFs
  - Chassis-to-chassis stateless failover (also known as "IPSec Stateless Failover Using HSRP")
  - Blade-to-Blade failover (also known as "IPSec Stateful Failover Using a Blade Failure Group")
  - 1024 TP tunnels
  - DMVPN (Cisco IOS Release 12.2(18)SXE and later)
  - More than one IPSec VPN SPA in a chassis

    Note Although more than one IPSec VPN SPA in a chassis is supported beginning with Cisco IOS Release 12.2(18)SXE, in VRF mode, there is no configuration difference between multiple IPSec VPN SPA operation and single IPSec VPN SPA operation. For multiple IPSec VPN SPA operation, the only change is to the output of the **show crypto vlan** command; an example follows:

    ```
    Interface Tu1 on IPSec Service Module port Gi7/1/1 connected to VRF vrf1
    Interface VLAN 2 on IPSec Service Module port Gi7/1/1 connected to VRF vrf2
    ```

  - The IPSec VPN SPA supports one or more outside interfaces (the exact number is determined by your system resources).
  - Inside VRFs (IVRFs), the VRFs on the unprotected (LAN) side, are supported.
  - As of Cisco IOS Release 12.2(33)SRA, front door VRFs (FVRFs) are now supported.
- Unsupported features in VRF mode are as follows:
  - Front door VRFs (FVRFs) are not supported in the Cisco IOS Releases 12.2(18)SX.

- Chassis-to-chassis stateful failover (also known as “IPSec Stateful Failover Using HSRP and SSP”) is not supported in Cisco IOS Release 12.2(33)SRA.
- CE-PE IPSec-only tunnels
- MPLS over GRE (tag switching on tunnel interfaces)
- PE-PE encryption (IPSec only) over MPLS
- PE-PE encryption (tunnel protection) over MPLS
- Multicast VPN (MVPN) over IPSec only

Note Multicast VPN is supported only to the extent that Cisco IOS supports it; multicast traffic is not accelerated by the IPSec VPN SPA. IPSec does not operate on multicast packets; if these packets go through the IPSec VPN SPA, they will be passed through.

- Non-IP version 4 traffic over TP tunnels

Note Non-IP version 4 packets are supported by Cisco IOS. IPSec does not operate on Non-IP version 4 packets; if these packets go through the IPSec VPN SPA, they will be passed through.

- Users may not apply an ACL to the ingress interface because it will interfere with the packet flow.

Note If an ACL is applied during the configuration of VRF mode, non-deterministic behavior will result.

- QoS support
- Policy-based routing (PBR)
- Path MTU discovery
- Unlike IPSec VPN SPA crypto-connect mode configurations, when configuring VPNs in VRF mode, you do not use the **crypto connect vlan** command.
- Secondary IP addresses on interfaces are not supported.
- The reverse route “remote peer” option is not supported.

Note The procedure for configuring a VPN in VRF mode varies based on whether you are using tunnel protection or not.

Configuring VPNs in VRF Mode with Crypto Maps

To configure a VPN in VRF mode on the IPSec VPN SPA with crypto maps, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **mls mpls tunnel-recir**	(Optional) Enables tunnel-MPLS recirculation. **Note** You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.
Step 2	Router(config)# **crypto engine mode vrf**	Enables VRF mode for the IPSec VPN SPA. **Note** After enabling or disabling VRF mode using the **crypto engine mode vrf** command, you must reload the supervisor engine.
Step 3	Router(config)# **ip vrf** *vrf-name*	Configures a VRF routing table and enters VRF configuration mode. • *vrf-name*—Name assigned to the VRF.
Step 4	Router(config-vrf)# **rd** *route-distinguisher*	Creates routing and forwarding tables for a VRF. • *route-distinguisher*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1).
Step 5	Router(config-vrf)# **route-target export** *route-target-ext-community*	Creates lists of export route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 6	Router(config-vrf)# **route-target import** *route-target-ext-community*	Creates lists of import route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 7	Router(config-vrf)# **exit**	Exits VRF configuration mode.

	Command	Purpose
Step 8	Router(config)# **crypto keyring** *keyring-name* [**vrf** *fvrf-name*]	Defines a crypto keyring to be used during IKE authentication and enters keyring configuration mode. • *keyring-name*—Name of the crypto keyring. • *fvrf-name*—(Optional) Front door virtual routing and forwarding (FVRF) name to which the keyring will be referenced. *fvrf-name* must match the FVRF name that was defined during virtual routing and forwarding (VRF) configuration.
Step 9	Router(config-keyring)# **pre-shared-key** {**address** *address* [*mask*] | **hostname** *hostname*} **key** *key*	Defines a preshared key to be used for IKE authentication. • *address* [*mask*]—IP address of the remote peer or a subnet and mask. • *hostname*—Fully qualified domain name of the peer. • *key*—Specifies the secret key.
Step 10	Router(config-keyring)# **exit**	Exits keyring configuration mode.
Step 11	Router(config)# **crypto ipsec transform-set** *transform-set-name transform1*[*transform2*[*transform3*]]	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPSec security protocols and algorithms. Accepted values are described in the *Cisco IOS Security Command Reference*.
Step 12	Router(config-crypto-trans)# **exit**	Exits crypto transform configuration mode.
Step 13	Router(config)# **crypto isakmp policy** *priority*	Defines an IKE policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest.
Step 14	Router(config-isakmp)# **authentication pre-share**	Specifies the authentication method with an IKE policy. • **pre-share**—Specifies preshared keys as the authentication method.
Step 15	Router(config-isakmp)# **lifetime** *seconds*	Specifies the lifetime of an IKE SA. • *seconds*—Number of seconds each SA should exist before expiring. Use an integer from 60 to 86,400 seconds. Default is 86,400 (one day).
Step 16	Router(config-isakmp)# **exit**	Exits ISAKMP policy configuration mode.

	Command	Purpose
Step 17	Router(config)# **crypto isakmp profile** *profile-name*	Defines an ISAKMP profile and enters ISAKMP profile configuration mode. • *profile-name*—Name of the user profile.
Step 18	Router(config-isa-prof)# **vrf** *ivrf*	Defines the VRF to which the IPSec tunnel will be mapped. • *ivrf*—Name of the VRF to which the IPSec tunnel will be mapped. Enter the same value specified in Step 3.
Step 19	Router(config-isa-prof)# **keyring** *keyring-name*	Configures a keyring within an ISAKMP profile. • *keyring-name*—Keyring name. This name must match the keyring name that was defined in global configuration. Enter the value specified in Step 8.
Step 20	Router(config-isa-prof)# **match identity address** *address* [*mask*] [*vrf*]	Matches an identity from a peer in an ISAKMP profile. • *address* [*mask*]—IP address of the remote peer or a subnet and mask. • [*vrf*]—(Optional) This argument is only required when configuring a front door VRF (FVRF). This argument specifies that the address is an FVRF instance.
Step 21	Router(config-isa-prof)# **exit**	Exits ISAKMP profile configuration mode.
Step 22	Router(config)# **access list** *access-list-number* {**deny** | **permit**} **ip host** *source* **host** *destination*	Defines an extended IP access list. • *access-list-number*—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699. • {**deny** | **permit**}—Denies or permits access if the conditions are met. • *source*—Number of the host from which the packet is being sent. • *destination*—Number of the host to which the packet is being sent.
Step 23	Router(config)# **crypto map** *map-name seq-number* **ipsec-isakmp**	Creates or modifies a crypto map entry and enters the crypto map configuration mode. • *map-name*—Name that identifies the crypto map set. • *seq-number*—Sequence number you assign to the crypto map entry. Lower values have higher priority. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec security associations.

	Command	Purpose
Step 24	Router(config-crypto-map)# **set peer** {*hostname* | *ip-address*}	Specifies an IPSec peer in a crypto map entry. • {*hostname* | *ip-address*}—IPSec peer host name or IP address. Enter the value specified in Step 20.
Step 25	Router(config-crypto-map)# **set transform-set** *transform-set-name*	Specifies which transform sets can be used with the crypto map entry. • *transform-set-name*—Name of the transform set. Enter the value specified in Step 11.
Step 26	Router(config-crypto-map)# **set isakmp-profile** *profile-name*	Sets the ISAKMP profile name. • *profile-name*—Name of the ISAKMP profile. Enter the value entered in Step 17.
Step 27	Router(config-crypto-map)# **match address** [*access-list-id* | *name*]	Specifies an extended access list for the crypto map entry. • *access-list-id*—Identifies the extended access list by its name or number. Enter the value specified in Step 22. • *name*—(Optional) Identifies the named encryption access list. This name should match the name argument of the named encryption access list being matched.
Step 28	Router(config-crypto-map)# **exit**	Exits crypto map configuration mode.
Step 29	Router(config)# **crypto map** *map-name* **local-address** *interface-id*	Specifies and names an identifying interface to be used by the crypto map for IPSec traffic. • *map-name*—Name that identifies the crypto map set. Enter the value specified in Step 23. • **local-address** *interface-id*—Name of interface that has the local address of the router.
Step 30	Router(config)# **interface fastethernet** *slot/port*	Configures a Fast Ethernet interface and enters interface configuration mode.
Step 31	Router(config-if)# **ip vrf forwarding** *vrf-name*	Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF. Enter the value specified in Step 3.
Step 32	Router(config-if)# **ip address** *address mask*	Sets a primary or secondary IP address for the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 33	Router(config-if)# **no shut**	Enables the interface.

	Command	Purpose
Step 34	Router(config-if)# **interface gigabitethernet** *slot/subslot/port*	Configures a Gigabit Ethernet interface. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA. Specify values for the *interface-id* specified in Step 29.
Step 35	Router(config-if)# **ip vrf forwarding** *vrf-name*	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.
Step 36	Router(config-if)# **ip address** *address mask*	Sets a primary or secondary IP address for an interface. • *address*—IP address. • *mask*—Subnet mask.
Step 37	Router(config-if)# **crypto engine slot** *slot/subslot* **outside**	Assigns the specified crypto engine to the interface. • *slot/subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 38	Router(config-if)# **no shut**	Enables the interface.
Step 39	Router(config-if)# **exit**	Exits interface configuration mode.
Step 40	Router(config)# **interface** *vlan-id*	Configures a VLAN interface and enters interface configuration mode. • *vlan-id*—VLAN identifier.
Step 41	Router(config-if)# **ip vrf forwarding** *vrf-name*	Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF. Enter the value specified in Step 3.
Step 42	Router(config-if)# **ip address** *address mask*	Sets a primary or secondary IP address for the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 43	Router(config-if)# **crypto map** *map-name*	Applies a previously defined crypto map set to an interface. • *map-name*—Name that identifies the crypto map set. Enter the value specified in Step 23.
Step 44	Router(config-if)# **crypto engine slot** *slot/subslot* **inside**	Assigns the specified crypto engine to the interface. • *slot/subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 45	Router(config-if)# **exit**	Exits interface configuration mode.

	Command	Purpose
Step 46	Router(config)# **ip route vrf** *vrf-name prefix mask interface-number*	Establishes static routes for a VRF. • *vrf-name*—Name of the VRF for the static route. Enter the value specified in Step 3. • *prefix*—IP route prefix for the destination, in dotted-decimal format. • *mask*—Prefix mask for the destination, in dotted decimal format. • *interface-number*—Number identifying the network interface to use. Enter the *vlan-id* value specified in Step 40.
Step 47	Router(config)# **end**	Returns to privileged EXEC mode.

For complete configuration information for VRF-Aware IPSec, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t15/ft_vrfip.htm

For configuration examples, see the "VRF Mode Using Crypto Maps Configuration Examples" section on page 23-64.

Configuring GRE Over IPSec

In addition to choosing to configure your VPN using crypto-connect mode or VRF mode, the following additional GRE configuration options are available:

Configuring GRE Tunneling in Crypto-Connect Mode

Generic Routing Encapsulation (GRE) is a tunneling protocol that can encapsulate a wide variety of protocol packet types inside IP tunnels, creating a virtual point-to-point link to routers at remote points over an IP network. When GRE is used in conjunction with IPSec, only tunnel mode can be used. Tunnel mode adds an IPSec header to the GRE packet.

Note The IPSec VPN SPA is able to accelerate packet processing for up to 2048 GRE tunnels per chassis; excess tunnels go through the Route Processor. The router supports any number of GRE tunnels, but adding more IPSec VPN SPAs does not increase the 2048 tunnels per-chassis maximum. If you configure more than 2048 tunnels per chassis, you could overload the Route Processor. Monitor the Route Processor CPU utilization when configuring more than 2048 tunnels per chassis.

GRE Tunneling Configuration Guidelines

Follow these guidelines when configuring GRE tunneling using the IPSec VPN SPA:

- In a Cisco 7600 series router, GRE encapsulation and decapsulation is traditionally performed by the Route Processor or the Supervisor Engine hardware. When routing indicates that encapsulated packets for a GRE tunnel will egress through an interface VLAN that is attached to an IPSec VPN SPA inside port, the IPSec VPN SPA attempts to take over the GRE tunnel interface only if the Supervisor Engine 720 is unable to process the GRE tunnel interface in hardware. If the Supervisor Engine 720 cannot process the GRE tunnel interface in hardware, the IPSec VPN SPA will determine if it can take over the interface. By seizing the tunnel, the IPSec VPN SPA takes the GRE encapsulation and decapsulation duty from the Route Processor. No explicit configuration changes are required to use this feature; configure GRE as you normally would. As long as routing sends the GRE-encapsulated packets out an interface VLAN, the IPSec VPN SPA will seize the GRE tunnel.
- The following are cases where the Supervisor Engine will not take over the tunnel but the IPSec VPN SPA will take over the tunnel if it meets the criteria discussed in the previous list item:
  - If the same source address is used for more than one GRE tunnel, the Supervisor Engine will only take over the first tunnel, but not subsequent tunnels.
  - If the HSRP virtual IP address is configured as the source address of the tunnel, the Supervisor Engine will not take over the tunnel.
- One VLAN is used for each GRE tunnel regardless of whether the IPSec VPN SPA takes over the tunnel.
- If routing information changes and the GRE-encapsulated packets no longer egress through an interface VLAN, the IPSec VPN SPA yields the GRE tunnel. After the IPSec VPN SPA yields the tunnel, the Route Processor resumes encapsulation and decapsulation, which increases CPU utilization on the Route Processor.

Caution Ensure that your GRE tunnel configuration does not overload the Route Processor.

- A delay (up to 10 seconds) occurs between routing changes and the IPSec VPN SPA seizing the GRE tunnel.
- Do not attach a crypto map set to a generic routing encapsulation (GRE) tunnel interface. Instead, attach the crypto map set to all of the ingress and egress interfaces over which the GRE tunnel spans.
- The crypto map must only be applied to the interface VLAN and not to the tunnel interface.
- HSRP/GRE is supported.
- Tunnel mode is the only GRE mode that is supported.
- The following options are not supported: checksum enabled, sequence check enabled, tunnel key, IP security options, IP policy, service policy, traffic shaping, QoS preclassification, NAT, and ACLs. If any of these options are specified, the IPSec VPN SPA will not seize the GRE tunnel.
- GRE tunneling of all non-IP packets is done by the Route Processor even if the tunnel is seized by the IPSec VPN SPA.

To configure a GRE tunnel in crypto-connect mode, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface tunnel** *number*	Creates the tunnel interface if it does not exist and enters interface configuration mode. • *number*—Number of the tunnel interface to be configured.
Step 2	Router(config-if)# **ip address** *address*	Sets the IP address of the tunnel interface. • *address*—IP address.
Step 3	Router(config-if)# **tunnel source** {*ip-address* | *type number*}	Configures the tunnel source. The source is the router where traffic is received from the customer network. • *ip-address*—IP address to use as the source address for packets in the tunnel. • *type number*—Interface type and number; for example, VLAN1.
Step 4	Router(config-if)# **tunnel destination** {*hostname* | *ip-address*}	Sets the IP address of the destination of the tunnel interface. The destination address is the router that transfers packets into the receiving customer network. • *hostname*—Name of the host destination. • *ip-address*—IP address of the host destination expressed in decimal in four-part, dotted notation.
Step 5	Router(config-if)# **exit**	Exits interface configuration mode.

Verifying the GRE Tunneling Configuration

To verify that the IPSec VPN SPA has seized the GRE tunnel, enter the **show crypto vlan** command:

```
Router# show crypto vlan

Interface VLAN 101 on IPSec Service Module port 7/1/1 connected to AT4/0/0.101
    Tunnel101 is accelerated via IPSec SM in sublot 7/1
Router#
```

For complete configuration information for GRE tunneling, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s17/12s_tos.htm

For GRE tunneling configuration examples, see the "GRE Tunneling in Crypto-Connect Mode Configuration Examples" section on page 23-83.

Configuring VPNs in VRF Mode with Tunnel Protection

Note Tunnel protection is supported only in VRF mode.

This section describes how to configure a VPN in VRF mode on the IPSec VPN SPA with tunnel protection. When you configure IPSec, a crypto map is attached to an interface to enable IPSec. With tunnel protection, there is no need for a crypto map or ACL to be attached to the interface. A crypto policy is attached directly to the tunnel interface. Any traffic routed by the interface is encapsulated in GRE and then encrypted using IPSec. The tunnel protection feature can be applied to point-to-point GRE.

VRF Mode with Tunnel Protection Configuration Guidelines and Restrictions

When configuring a VPN in VRF mode on the IPSec VPN SPA with tunnel protection, follow these guidelines and restrictions:

- Do not configure any options (such as sequence numbers or tunnel keys) that prevent the IPSec VPN SPA from seizing the GRE tunnel.
- Do not configure the GRE tunnel keepalive feature.
- The **ip vrf forwarding** command is no longer required when configuring GRE with tunnel protection.

To configure a VPN in VRF mode using tunnel protection, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **mls mpls tunnel-recir**	(Optional) Enables tunnel-MPLS recirculation. **Note** You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.
Step 2	Router(config)# **crypto engine mode vrf**	Enables VRF mode for the IPSec VPN SPA. **Note** After enabling or disabling VRF mode using the **crypto engine mode vrf** command, you must reload the supervisor engine.
Step 3	Router(config)# **ip vrf** *vrf-name*	Configures a VRF routing table and enters VRF configuration mode. • *vrf-name*—Name assigned to the VRF.
Step 4	Router(config-vrf)# **rd** *route-distinguisher*	Creates routing and forwarding tables for a VRF. • *route-distinguisher*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1).
Step 5	Router(config-vrf)# **route-target export** *route-target-ext-community*	Creates lists of export route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.

	Command	Purpose
Step 6	Router(config-vrf)# **route-target import** *route-target-ext-community*	Creates lists of import route-target extended communities for the specified VRF. • *route-target-ext-community*—Specifies an autonomous system number (ASN) and an arbitrary number (for example, 101:3) or an IP address and an arbitrary number (for example, 192.168.122.15:1). Enter the *route-distinguisher* value specified in Step 4.
Step 7	Router(config-vrf)# **exit**	Exits VRF configuration mode.
Step 8	Router(config)# **crypto keyring** *keyring-name* [**vrf** *fvrf-name*]	Defines a crypto keyring to be used during IKE authentication and enters keyring configuration mode. • *keyring-name*—Name of the crypto keyring. • *fvrf-name*—(Optional) Front door virtual routing and forwarding (FVRF) name to which the keyring will be referenced. *fvrf-name* must match the FVRF name that was defined during virtual routing and forwarding (VRF) configuration.
Step 9	Router(config-keyring)# **pre-shared-key** {**address** *address* [*mask*] | **hostname** *hostname*} **key** *key*	Defines a preshared key to be used for IKE authentication. • *address* [*mask*]—IP address of the remote peer or a subnet and mask. • *hostname*—Fully qualified domain name of the peer. • *key*—Specifies the secret key.
Step 10	Router(config-keyring)# **exit**	Exits keyring configuration mode.
Step 11	Router(config)# **crypto ipsec transform-set** *transform-set-name transform1*[*transform2*[*transform3*]]	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPSec security protocols and algorithms. Accepted values are described in the *Cisco IOS Security Command Reference*.
Step 12	Router(config-crypto-trans)# **exit**	Exits crypto transform configuration mode.
Step 13	Router(config)# **crypto isakmp policy** *priority*	Defines an IKE policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest.

	Command	Purpose
Step 14	Router(config-isakmp)# **authentication pre-share**	Specifies the authentication method with an IKE policy. • **pre-share**—Specifies preshared keys as the authentication method.
Step 15	Router(config-isakmp)# **lifetime** *seconds*	Specifies the lifetime of an IKE SA. • *seconds*—Number of seconds each SA should exist before expiring. Use an integer from 60 to 86,400 seconds. Default is 86,400 (one day).
Step 16	Router(config-isakmp)# **exit**	Exits ISAKMP policy configuration mode.
Step 17	Router(config)# **crypto isakmp profile** *profile-name*	Defines an ISAKMP profile and enters ISAKMP profile configuration mode. • *profile-name*—Name of the user profile.
Step 18	Router(config-isa-prof)# **keyring** *keyring-name*	Configures a keyring within an ISAKMP profile. • *keyring-name*—Keyring name. This name must match the keyring name that was defined in global configuration. Enter the value specified in Step 8.
Step 19	Router(config-isa-prof)# **match identity address** *address* [*mask*] [*vrf*]	Matches an identity from a peer in an ISAKMP profile. • *address* [*mask*]—IP address of the remote peer or a subnet and mask. • [*vrf*]—(Optional) This argument is only required when configuring a front door virtual routing and forwarding (FVRF). This argument specifies that the address is an FVRF instance.
Step 20	Router(config-isa-prof)# **exit**	Exits ISAKMP profile configuration mode.
Step 21	Router(config)# **access list** *access-list-number* {**deny** | **permit**} **ip host** *source* **host** *destination*	Defines an extended IP access list. • *access-list-number*—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699. • {**deny** | **permit**}—Denies or permits access if the conditions are met. • *source*—Number of the host from which the packet is being sent. • *destination*—Number of the host to which the packet is being sent.
Step 22	Router(config)# **crypto ipsec profile** *profile-name*	Defines an IPSec profile and enters IPSec profile configuration mode. • *profile-name*—Name of the user profile.
Step 23	Router(config-ipsec-profile)# **set transform-set** *transform-set-name*	Specifies which transform sets can be used with the crypto map entry. • *transform-set-name*—Name of the transform set. Enter the value specified in Step 11.

	Command	Purpose
Step 24	Router(config-ipsec-profile)# **set isakmp-profile** *profile-name*	Sets the ISAKMP profile name. • *profile-name*—Name of the ISAKMP profile. Enter the value entered in Step 17.
Step 25	Router(config-ipsec-profile)# **exit**	Exits IPSec profile configuration mode.
Step 26	Router(config)# **interface** *interface-name*	Configures a tunnel interface and enters interface configuration mode. • *interface-name*—Name assigned to the interface.
Step 27	Router(config-if)# **ip vrf forwarding** *vrf-name*	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF. Enter the value specified in Step 3.
Step 28	Router(config-if)# **ip address** *address mask*	Sets a primary or secondary IP address for the interface. • *address*—IP address. • *mask*—Subnet mask.
Step 29	Router(config-if)# **tunnel source** *ip-address*	Sets the source address of a tunnel interface. • *ip-address*—IP address to use as the source address for packets in the tunnel.
Step 30	Router(config-if)# **tunnel vrf** *vrf-name*	(Optional) Associates a VPN routing and forwarding instance (VRF) with a specific tunnel destination, interface or subinterface. This step is only required when configuring a front door VRF (FVRF). • *vrf-name*—Name assigned to the VRF.
Step 31	Router(config-if)# **tunnel destination** *ip-address*	Sets the destination address of a tunnel interface. • *ip-address*—IP address to use as the destination address for packets in the tunnel.
Step 32	Router(config-if)# **tunnel protection ipsec** *crypto-policy-name*	Associates a tunnel interface with an IPSec profile. • *crypto-policy-name*—Enter the value specified in Step 22.
Step 33	Router(config-if)# **crypto engine slot** *slot/subslot* **inside**	Assigns the specified crypto engine to the interface. • *slot/subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 34	Router(config-if)# **interface fastethernet** *slot/subslot*	Configures a Fast Ethernet interface.
Step 35	Router(config-if)# **ip vrf forwarding** *vrf-name*	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.

	Command	Purpose
Step 36	Router(config-if)# **ip address** *address mask*	Sets a primary or secondary IP address for an interface. • *address*—IP address. • *mask*—Subnet mask.
Step 37	Router(config-if)# **no shut**	Enables the interface.
Step 38	Router(config-if)# **interface gigabitethernet** *slot/subslot/port*	Configures a Gigabit Ethernet interface. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 39	Router(config-if)# **ip vrf forwarding** *vrf-name*	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.
Step 40	Router(config-if)# **ip address** *address mask*	Sets a primary or secondary IP address for an interface. • *address*—IP address. Enter the value specified in Step 30. • *mask*—Subnet mask.
Step 41	Router(config-if)# **crypto engine slot** *slot/subslot* **outside**	Assigns the specified crypto engine to the interface. • *slot/subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 42	Router(config-if)# **no shut**	Enables the interface.
Step 43	Router(config-if)# **exit**	Exits interface configuration mode.

For a configuration example, see the "VRF Mode with Tunnel Protection Configuration Example" section on page 23-85.

Configuring the GRE Takeover Criteria

You can configure the takeover criteria for Generic Routing Encapsulation (GRE) processing by using the **crypto engine gre supervisor** or **crypto engine gre vpnblade** commands. These two commands allow you to specify whether the GRE processing should be done by the Supervisor Engine hardware or the Route Processor or the IPSec VPN SPA.

Note The GRE takeover criteria commands are supported only in Cisco IOS Release 12.2(18)SXE5 and later.

To configure a router to process GRE using the Supervisor Engine hardware or the Route Processor (RP), use the **crypto engine gre supervisor** command. When this command is specified, GRE processing by the Supervisor Engine hardware takes precedence over processing by the RP (unless the tunnels are from duplicate sources); the RP only takes over GRE processing if the Supervisor Engine hardware cannot do the processing. If this command is configured, duplicate source GREs will be processed by the RP.

To configure a router to process Generic Routing Encapsulation (GRE) using the IPSec VPN SPA, use the **crypto engine gre vpnblade** command. If the IPSec VPN SPA cannot take over the GRE processing, the GRE processing will be handled either by Supervisor Engine hardware (which has precedence) or the Route Processor (RP).

Both of these commands can be configured globally or at an individual tunnel.

Individual tunnel configuration takes precedence over the global configuration. For example, when the **crypto engine gre supervisor** command is configured at the global configuration level, the command will apply to all tunnels except those tunnels that have been configured individually using either a **crypto engine gre supervisor** command or a **crypto engine gre vpnblade** command.

At any time, only one of the two commands (**crypto engine gre supervisor** or **crypto engine gre vpnblade**) can be configured globally or individually at a tunnel. If either command is already configured, configuring the second command will overwrite the first command, and only the configuration applied by the second command will be used.

GRE Takeover Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring GRE takeover on the IPSec VPN SPA:

- For a GRE tunnel to be taken over by the IPSec VPN SPA, it must first satisfy the following criteria:
  - The GRE tunnel interface must be up.
  - The route to the tunnel destination must go through the IPSec VPN SPA.
  - The Address Resolution Protocol (ARP) entry for the next hop must exist.
  - The tunnel mode must be GRE.
  - The only supported options are **tunnel ttl**, **tunnel tos**, and **keepalive**. If any of the following options are configured, then the tunnel will not be taken over:
    - **tunnel key**
    - **tunnel sequence-datagrams**
    - **tunnel checksum**

    All other options configured are ignored.
- If the GRE tunnels have the same source and destination addresses, then the IPSec VPN SPA will, at most, take over only one of them, and the determination of which specific tunnel is taken over is random.
- GRE processing cannot be performed by the IPSec VPN SPA if any of the following options is configured on the tunnel interface:
  - DMVPN
  - IP policy
  - NAT
  - Service policy
  - Traffic shaping

- QoS
- ACL

- In crypto-connect mode, the IPSec VPN SPA will not take over GRE processing when the interface VLAN has no crypto map attached. The crypto map must be applied to the interface VLAN and not to the tunnel interface.
- If the IPSec VPN SPA cannot take over the GRE processing, the GRE processing will be handled either by the Supervisor Engine hardware (which has precedence) or the RP.
- When neither the **crypto engine gre supervisor** command, nor the **crypto engine gre vpnblade** command is specified globally or individually for a tunnel, GRE processing will be performed based on the following IPSec VPN SPA GRE takeover criteria:
  - The Supervisor Engine hardware does not take over GRE processing.
  - Protocol Independent Multicast (PIM) is configured on the tunnel.
  - The tunnels are from duplicate tunnel sources and more than one tunnel is up. (If only one tunnel is up, the Supervisor Engine hardware can still perform the GRE processing.)
- When a new configuration file is copied to the running configuration, the new configuration will overwrite the old configuration for the **crypto engine gre vpnblade** and **crypto engine gre supervisor** commands. If the new configuration does not specify a GRE takeover criteria globally or for an individual tunnel, the existing old configuration will be used.

Configuring the GRE Takeover Criteria Globally

To configure the GRE takeover criteria globally (so that it affects all tunnels except those tunnels that have been configured individually using either a **crypto engine gre supervisor** command or a **crypto engine gre vpnblade** command), perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto engine gre supervisor** or Router(config)# **crypto engine gre vpnblad**e	Configures a router to process GRE using the Supervisor Engine hardware or the Route Processor (RP). Configures a router to process GRE using the IPSec VPN SPA.

Configuring the GRE Takeover Criteria at an Individual Tunnel

To configure the GRE takeover criteria at an individual tunnel (so that it affects only a specific tunnel), perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface tunnel** *number*	Creates the tunnel interface if it does not exist and enters interface configuration mode. • *number*—Number of the tunnel interface to be configured.
Step 2	Router(config-if)# **crypto engine gre supervisor** or Router(config-if)# **crypto engine gre vpnblade**	Configures a router to process GRE using the Supervisor Engine hardware or the Route Processor (RP). or Configures a router to process GRE using the IPSec VPN SPA.

For GRE takeover criteria configuration examples, see the "GRE Takeover Criteria Configuration Examples" section on page 23-88.

Configuring IP Multicast Over a GRE Tunnel

IP multicast is a bandwidth-conserving technology that reduces traffic by simultaneously delivering a single stream of information to multiple recipients. GRE is a tunneling protocol developed by Cisco and commonly used with IPSec that encapsulates a wide variety of protocol packet types inside IP tunnels, creating a virtual point-to-point link to Cisco routers at remote points over an IP network.

Note IP Multicast Over a GRE Tunnel is only supported in Cisco IOS Release 12.2(18)SXF2 and later.

In some network scenarios, you might want to configure your network to use GRE tunnels to send Protocol Independent Multicast (PIM) and multicast traffic between routers. Typically, this occurs when the multicast source and receiver are separated by an IP cloud that is not configured for IP multicast routing. In such network scenarios, configuring a tunnel across an IP cloud with PIM enabled transports multicast packets toward the receiver. The configuration of IP multicast over a GRE tunnel using the IPSec VPN SPA involves three key steps:

- Configuring single-SPA mode for multicast traffic
- Configuring multicast globally
- Configuring PIM at the tunnel interfaces

IP Multicast Over a GRE Tunnel Configuration Guidelines and Restrictions

Follow these guidelines when configuring IP multicast over a GRE tunnel:

- When the **hw-module slot subslot only** command is executed, it automatically takes a reset action on the Cisco 7600 SSC-400 and issues the following prompt to the console:

```
Module n will be reset? Confirm [n]:
```

The prompt will default to “N” (no). You must type “Y” (yes) to activate the reset action.

- When in single-SPA mode, if you manually plug in a second SPA, or if you attempt to reset the SPA (by entering a **no hw-module subslot shutdown** command, for example), a message is displayed on the router console that refers you to the customer documentation.
- If PIM is configured, and the GRE tunnel interface satisfies the rest of the tunnel takeover criteria, the GRE processing of the multicast packets is taken over by the IPSec VPN SPA.
- GRE processing of IP multicast packets is taken over by the IPSec VPN SPA if the GRE tunnel interface satisfies the following tunnel takeover criteria:
  - The tunnel is up.
  - The are no other tunnels with the same source destination pair.
  - Tunnel protection has not been applied to the tunnel interface in crypto-connect mode.
  - Tunnel protection has been applied to the tunnel interface in VRF mode.
  - The tunnel is not a multipoint GRE tunnel.
  - PIM is configured on the tunnel.
  - None of the following features are configured on the tunnel: **tunnel key**, **tunnel sequence-datagrams**, **tunnel checksum**, **tunnel udlr address-resolution**, **tunnel udlr** r**eceive-only**, **tunnel udlr send-only**, **ip proxy-mobile tunnel reverse**, **ip policy**, service policy, traffic shaping, QoS pre-classification, NAT, or ACLs. If any of these options are specified, the IPSec VPN SPA will not seize the GRE tunnel.
- When a tunnel is configured for multicast traffic, the **crypto engine gre supervisor** command should not be applied to the tunnel.

Configuring Single-SPA Mode for IP Multicast Traffic

Before you configure IP multicast on the IPSec VPN SPA, it is strongly recommended that you change the mode of the Cisco 7600 SSC-400 to allocate full buffers to the specified subslot using the **hw-module slot subslot only** command. If this command is not used, the total amount of buffers available is divided between the two subslots on the Cisco 7600 SSC-400.

To allocate full buffers to the specified subslot, use the **hw-module slot subslot only** command as follows beginning in global configuration mode:

Router(config)# **hw-module slot** *slot* **subslot** *subslot* **only**

In this command:

- *slot* specifies the slot where the Cisco 7600 SSC-400 is located.
- *subslot* specifies the subslot where the IPSec VPN SPA is located.

Configuring IP Multicast Globally

You must enable IP multicast routing globally before you can enable PIM on the router interfaces.

To enable IP multicast routing globally, use the **ip multicast-routing** command as follows beginning in global configuration mode:

Router(config)# **ip multicast-routing**

Configuring PIM at the Tunnel Interfaces

You must enable PIM on all participating router interfaces before IP multicast will function.

To enable PIM, use the **ip pim** command as follows beginning in interface configuration mode:

Router(config-if)# **ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**}

In this command:

- **dense-mode** enables dense mode of operation.
- **sparse-mode** enables sparse mode of operation.
- **sparse-dense-mode** enables the interface in either sparse mode or dense mode of operation, depending on which mode the multicast group operates in.

Verifying the IP Multicast Over a GRE Tunnel Configuration

To verify the IP multicast over a GRE tunnel configuration, enter the **show crypto vlan** and **show ip mroute** commands.

Enter the **show crypto vlan** command to verify that the tunnel has been taken over by the IPSec VPN SPA:

```
Router(config)# show crypto vlan

Interface VLAN 100 on IPSec Service Module port Gi7/0/1 connected to Po1 with crypto map
set map_t3
Tunnel15 is accelerated via IPSec SM in subslot 7/0
```

Enter the **show ip mroute** command and look for the "H" flag to verify that the IP multicast traffic is hardware-switched:

```
Router# show ip mroute 230.1.1.5

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, M - MSDP created entry,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report, Z - Multicast Tunnel
Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 230.1.1.5), 01:23:45/00:03:16, RP 15.15.1.1, flags: SJC
Incoming interface: Null, RPF nbr 0.0.0.0
Outgoing interface list:
Tunnel15, Forward/Sparse-Dense, 00:25:47/00:03:16
(120.1.0.3, 230.1.1.5), 01:23:46/00:03:25, flags: T
Incoming interface: GigabitEthernet8/1, RPF nbr 0.0.0.0, RPF-MFD
Outgoing interface list:
Tunnel15, Forward/Sparse-Dense, 00:25:47/00:03:16, H
```

For IP multicast over GRE tunnels configuration examples, see the "IP Multicast Over a GRE Tunnel Configuration Example" section on page 23-88.

Configuring an IPSec Virtual Tunnel Interface

The IPSec Virtual Tunnel Interface (VTI) provides a routable interface type for terminating IPSec tunnels that greatly simplifies the configuration process when you need to provide protection for remote access, and provides a simpler alternative to using GRE tunnels and crypto maps with IPSec. In addition, the IPSec VTI simplifies network management and load balancing.

Note IPSec VTI is only supported as of Cisco IOS Release 12.2(33)SRA, and is only supported in VRF mode.

Note the following details about IPSec VTI routing and traffic encryption:

- You can enable routing protocols on the tunnel interface so that routing information can be propagated over the virtual tunnel. The router can establish neighbor relationships over the virtual tunnel interface. Interoperability with standard-based IPSec installations is possible through the use of the IP ANY ANY proxy. The static IPSec interface will negotiate and accept IP ANY ANY proxies.
- The IPSec VTI supports native IPSec tunneling and exhibits most of the properties of a physical interface.
- In the IPSec VTI, encryption occurs in the tunnel. Traffic is encrypted when it is forwarded to the tunnel interface. Traffic forwarding is handled by the IP routing table, and dynamic or static IP routing can be used to route the traffic to the virtual tunnel interface. Using IP routing to forward the traffic to encryption simplifies the IPSec VPN configuration because the use of ACLs with a crypto map in native IPSec configurations is not required. When IPSec VTIs are used, you can separate applications of NAT, ACLs, and QoS, and apply them to clear text or encrypted text, or both. When crypto maps are used, there is no easy way to specify forced encryption features.

IPSec Virtual Tunnel Interface Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring IPSec VTI:

- Only static VTI is currently supported.
- IPSec stateful failover is not supported with IPSec VTIs.
- Only strict IP ANY ANY proxy is supported.
- The IPSec transform set must be configured only in tunnel mode.
- The IKE security association (SA) is bound to the virtual tunnel interface. Because it is bound to the virtual tunnel interface, the same IKE SA cannot be used for a crypto map.
- The IPSec virtual tunnel interface is limited to IP unicast, as opposed to GRE tunnels, which have a wider application for IPSec implementation.
- Multicast over VTI is not supported.
- MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.
- If the packets are from MPLS, VTI cannot be the outgoing interface.

Configuring an IPSec Static Tunnel

To configure a static IPSec virtual tunnel interface, perform the following steps beginning in global configuration mode:

Step 1	Router(config)# **crypto ipsec profile** *profile-name*	Defines an IPSec profile and enters IPSec profile configuration mode. The IPSec profile defines the IP Security (IPSec) parameters that are to be used for IPSec encryption between two IPSec routers. • *profile-name*—Name of the user profile.
Step 2	Router(config-ipsec-profile)# **set transform-set** *transform-set-name* [*transform-set-name2 ...transform-set-name6*]	Specifies which transform sets can be used with the crypto map entry. • *transform-set-name*—Name of the transform set.
Step 3	Router(config)# **interface** *type slot*/[*subslot*]/*port*	Configures an interface type. • *type*—Type of interface being configured. • *slot*/[*subslot*]/ *port*—Number of the slot, subslot (optional), and port to be configured.
Step 4	Router(config-if)# **ip vrf forwarding** *vrf-name*	(Optional) Associates a VRF with an interface or subinterface. • *vrf-name*—Name assigned to the VRF.
Step 5	Router(config-if)# **ip address** *address mask*	Sets a primary or secondary IP address for an interface. • *address*—IP address. • *mask*—Subnet mask.
Step 6	Router(config-if)# **tunnel mode ipsec ipv4**	Defines the mode for the tunnel as IPSec and the transport as IPv4.
Step 7	Router(config-if)# **tunnel source** *ip-address*	Sets the source address of a tunnel interface. • *ip-address*—IP address to use as the source address for packets in the tunnel.
Step 8	Router(config-if)# **tunnel destination** *ip-address*	Sets the destination address of a tunnel interface. • *ip-address*—IP address to use as the destination address for packets in the tunnel.
Step 9	Router(config-if)# **tunnel vrf** *vrf-name*	(Optional) Associates a VPN routing and forwarding instance (VRF) with a specific tunnel destination. This step is only required when configuring a front door VRF (FVRF). • *vrf-name*—Name assigned to the VRF.

Step 10	Router(config-if)# **crypto engine slot** *slot/subslot* **inside**	Assigns the specified crypto engine to the interface. • *slot/subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 11	Router(config-if)# **tunnel protection ipsec profile** *name* [**shared**]	Associates a tunnel interface with an IPSec profile. • *name*—Name of the IPSec profile; this value must match the name specified in the **crypto ipsec profile** command in Step 1. • **shared**—(Optional) Allows the tunnel protection IPSec Security Association Database (SADB) to share the same dynamic crypto map instead of creating a unique crypto map per tunnel interface.

Verifying the IPSec Virtual Tunnel Interface Configuration

To confirm that your IPSec virtual tunnel interface configuration is working properly, enter the **show interfaces tunnel**, **show crypto session**, and **show ip route** commands.

The **show interfaces tunnel** command displays tunnel interface information, the **show crypto session** command displays status information for active crypto sessions, and the **show ip route** command displays the current state of the routing table.

Notice that in this display the Tunnel 0 is "up" and the line protocol is "up." If the line protocol is "down," the session is not active.

```
Router1# show interfaces tunnel 0

Tunnel0 is up, line protocol is up
Hardware is Tunnel
Internet address is 10.0.51.203/24
MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec,
reliability 255/255, txload 103/255, rxload 110/255
Encapsulation TUNNEL, loopback not set
Keepalive not set
Tunnel source 10.0.149.203, destination 10.0.149.217
Tunnel protocol/transport IPSEC/IP, key disabled, sequencing disabled
Tunnel TTL 255
Checksumming of packets disabled, fast tunneling enabled
Tunnel transmit bandwidth 8000 (kbps)
Tunnel receive bandwidth 8000 (kbps)
Tunnel protection via IPSec (profile "P1")
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 1/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/0 (size/max)
30 second input rate 13000 bits/sec, 34 packets/sec
30 second output rate 36000 bits/sec, 34 packets/sec
191320 packets input, 30129126 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
59968 packets output, 15369696 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 0 output buffers swapped out

Router1# show crypto session
Crypto session current status
```

```
Interface: Tunnel0
Session status: UP-ACTIVE
Peer: 10.0.149.217 port 500
IKE SA: local 10.0.149.203/500 remote 10.0.149.217/500 Active
IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
Active SAs: 4, origin: crypto map

Router1# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C 10.0.35.0/24 is directly connected, Ethernet3/3
S 10.0.36.0/24 is directly connected, Tunnel0
C 10.0.51.0/24 is directly connected, Tunnel0
C 10.0.149.0/24 is directly connected, Ethernet3/0
```

For more complete information about IPSec Virtual Tunnel Interface, refer to the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a008041faef.html

For IPSec Virtual Tunnel Interface configuration examples, see the "IPSec Virtual Tunnel Interfaces Configuration Example" section on page 23-90.

Configuring VPNs in Crypto Connect Alternative Mode

Crypto connect alternative (CCA) mode allows you to configure IPSec VTI without having to configure VRFs. Although CCA requires that VRF mode be configured globally using the **crypto engine mode vrf** command, tunnels are terminated in the global context rather than in VRFs. CCA is introduced in Cisco IOS Release 12.2(33)SRA.

The configuration steps for CCA are similar to the steps for IPSec VTI shown in the "Configuring an IPSec Static Tunnel" section on page 23-47 with the exception that the **ip vrf forwarding** *vrf-name* command and the **tunnel vrf** *vrf-name* command are not required.

For an example of IPSec Virtual Tunnel Interface configuration using CCA, see the "IPSec Virtual Tunnel Interfaces Configuration Example" section on page 23-90.

Configuration Examples

This section provides examples of the following configurations:

Access Port (Crypto-Connect Mode) Configuration Examples

This section provides examples of the access port configurations for the configuration shown in Figure 23-5 on page 23-52:

Note Switch port connections are not supported when a Cisco 7600 SIP-400 is present in the chassis.

In the following examples, the IPSec VPN SPA is installed in slot 5, subslot 1 (Gigabit Ethernet interfaces 5/1/1 and 5/1/2).

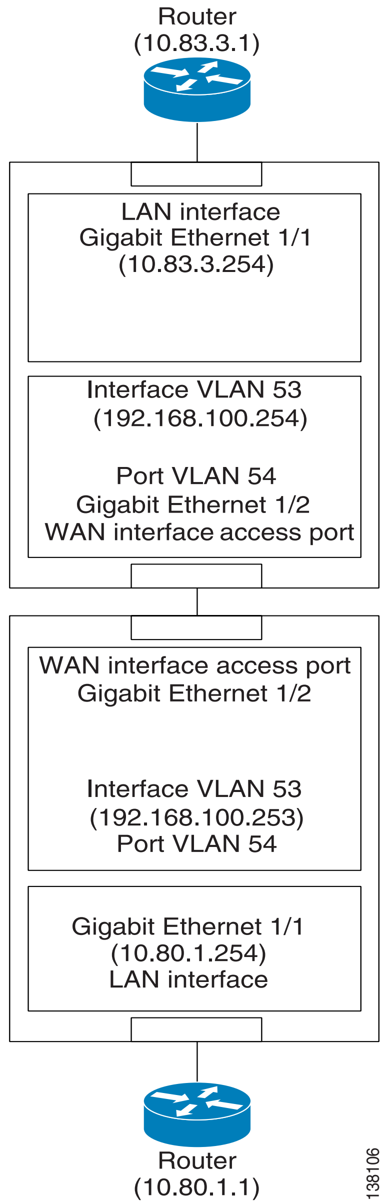
Access Port (Router 1) Configuration Example

The router 1 (access port) configuration for the configuration shown in Figure 23-5 is as follows:

```
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router-1
!
redundancy
 main-cpu
  auto-sync standard
diagnostic level complete
ip subnet-zero
!
no ip domain-lookup
!
ip ssh time-out 120
ip ssh authentication-retries 3
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key Jolly-Good-Fellow address 192.168.100.254
!
crypto ipsec transform-set TS-101 esp-3des esp-sha-hmac
!
crypto map MAP-101 10 ipsec-isakmp
 set peer 192.168.100.254
 set security-association lifetime kilobytes 10000
 set security-association lifetime seconds 86000
```

```
 set transform-set TS-101
 match address AEO-101
!
no spanning-tree vlan 53
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan53
 ip address 192.168.100.253 255.255.255.0
 crypto map MAP-101
 crypto engine slot 5/1
 !
interface Vlan54
 no ip address
 crypto connect vlan 53
!
interface GigabitEthernet1/1
 ip address 10.80.1.254 255.255.255.0
!
interface GigabitEthernet1/2
 switchport
 switchport access vlan 54
 switchport mode access
 no ip address
!
interface GigabitEthernet5/1/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,53,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
interface GigabitEthernet5/1/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,54,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
ip classless
ip route 10.83.3.0 255.255.255.0 192.168.100.254
no ip http server
!
ip access-list extended AEO-101
 permit ip 10.80.0.0 0.0.255.255 10.83.0.0 0.0.255.255
!
line con 0
line vty 0 4
 login
!
end
```

Figure 23-5   Access Port Configuration Example



Access Port (Router 2) Configuration Example

The router 2 (access port) configuration for the configuration shown in Figure 23-5 is as follows:

```
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
```

```
hostname Router-2
!
redundancy
 main-cpu
  auto-sync standard
diagnostic level complete
ip subnet-zero
!
no ip domain-lookup
!
ip ssh time-out 120
ip ssh authentication-retries 3
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key Jolly-Good-Fellow address 192.168.100.253
!
crypto ipsec transform-set TS-101 esp-3des esp-sha-hmac
!
crypto map MAP-101 10 ipsec-isakmp
 set peer 192.168.100.253
 set security-association lifetime kilobytes 10000
 set security-association lifetime seconds 86000
 set transform-set TS-101
 match address AEO-101
!
no spanning-tree vlan 53
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan53
 ip address 192.168.100.254 255.255.255.0
 crypto map MAP-101
 crypto engine slot 5/1
!
interface Vlan54
 no ip address
 crypto connect vlan 53
!!
interface GigabitEthernet1/1
 ip address 10.83.3.254 255.255.255.0
!
interface GigabitEthernet1/2
 switchport
 switchport access vlan 54
 switchport mode access
 no ip address
!
interface GigabitEthernet5/1/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,53,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
interface GigabitEthernet5/1/2
 switchport
 switchport trunk encapsulation dot1q
```

```
 switchport trunk allowed vlan 1,54,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
ip classless
ip route 10.80.1.0 255.255.255.0 192.168.100.253
no ip http server
!
ip access-list extended AEO-101
 permit ip 10.83.0.0 0.0.255.255 10.80.0.0 0.0.255.255
!
line con 0
line vty 0
 login
!
end
```

Routed Port (Crypto-Connect Mode) Configuration Examples

This section provides examples of the routed port configurations for the configuration shown in Figure 23-6 on page 23-56:

- Routed Port (Router 1) Configuration Example, page 23-54
- Routed Port (Router 2) Configuration Example, page 23-57

Note Switch port connections are not supported when a Cisco 7600 SIP-400 is present in the chassis.

In the following examples, the IPSec VPN SPA is installed in slot 5, subslot 1 (Gigabit Ethernet interfaces 5/1/1 and 5/1/2).

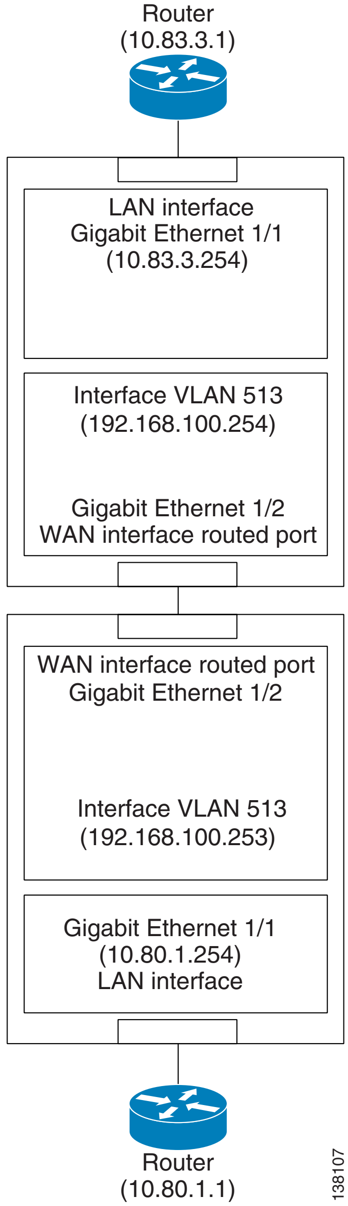
Routed Port (Router 1) Configuration Example

The router 1 (routed port) configuration for the configuration shown in Figure 23-6 is as follows:

```
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router-1
!
redundancy
 main-cpu
  auto-sync standard
diagnostic level complete
ip subnet-zero
!
no ip domain-lookup
!
ip ssh time-out 120
ip ssh authentication-retries 3
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
```

```
 group 2
crypto isakmp key Jolly-Good-Fellow address 192.168.100.254
!
crypto ipsec transform-set TS-101 esp-3des esp-sha-hmac
!
crypto map MAP-101 10 ipsec-isakmp
 set peer 192.168.100.254
 set security-association lifetime kilobytes 10000
 set security-association lifetime seconds 86000
 set transform-set TS-101
 match address AEO-101
!
!interface Vlan1
 no ip address
 shutdown
!
interface Vlan513
 ip address 192.168.100.253 255.255.255.0
 crypto map MAP-101
  crypto engine slot 5/1
  !
interface GigabitEthernet1/1
 ip address 10.80.1.254 255.255.255.0
!
interface GigabitEthernet1/2
 no ip address
 crypto connect vlan 513
!
interface GigabitEthernet5/1/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,513,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
interface GigabitEthernet5/1/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
ip classless
ip route 10.83.3.0 255.255.255.0 192.168.100.254
no ip http server
!
ip access-list extended AEO-101
 permit ip 10.80.0.0 0.0.255.255 10.83.0.0 0.0.255.255
!
line con 0
line vty 0 4
 login
!
end
```

Figure 23-6 Routed Port Configuration Example



Routed Port (Router 2) Configuration Example

The router 2 (routed port) configuration for the configuration shown in Figure 23-6 is as follows:

```
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router-2
!
redundancy
 main-cpu
  auto-sync standard
diagnostic level complete
ip subnet-zero
!
no ip domain-lookup
!
ip ssh time-out 120
ip ssh authentication-retries 3
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key Jolly-Good-Fellow address 192.168.100.253
!
crypto ipsec transform-set TS-101 esp-3des esp-sha-hmac
!
crypto map MAP-101 10 ipsec-isakmp
 set peer 192.168.100.253
 set security-association lifetime kilobytes 10000
 set security-association lifetime seconds 86000
 set transform-set TS-101
 match address AEO-101
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan513
 ip address 192.168.100.254 255.255.255.0
 crypto map MAP-101
crypto engine slot 5/1
!
interface GigabitEthernet1/1
 ip address 10.83.3.254 255.255.255.0
!
interface GigabitEthernet1/2
 no ip address
 crypto connect vlan 513
!
```

```
interface GigabitEthernet5/1/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,513,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
interface GigabitEthernet5/1/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
ip classless
ip route 10.80.1.0 255.255.255.0 192.168.100.253
no ip http server
!
ip access-list extended AEO-101
 permit ip 10.83.0.0 0.0.255.255 10.80.0.0 0.0.255.255
!
line con 0
line vty 0 4
 login
!
end
```

Trunk Port (Crypto-Connect Mode) Configuration Examples

This section provides examples of the trunk port configurations for the configuration shown in Figure 23-7 on page 23-60:

- Trunk Port (Router 1) Configuration Example, page 23-58
- Trunk Port (Router 2) Configuration Example, page 23-61

Note Switch port connections are not supported when a Cisco 7600 SIP-400 is present in the chassis.

In the following examples, the IPSec VPN SPA is installed in slot 5, subslot 1 (Gigabit Ethernet interfaces 5/1/1 and 5/1/2).

Trunk Port (Router 1) Configuration Example

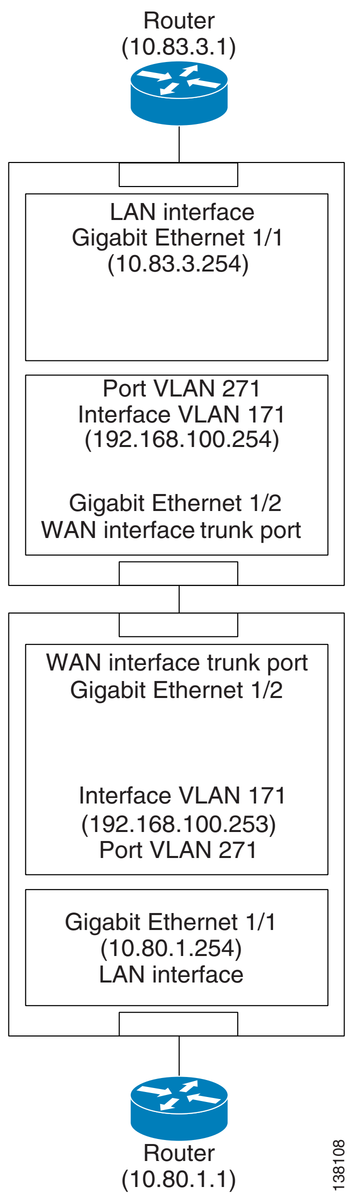
The router 1 (trunk port) configuration for the configuration shown in Figure 23-7 is as follows:

```
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router-1
!
redundancy
 main-cpu
  auto-sync standard
diagnostic level complete
```

```
ip subnet-zero
!
no ip domain-lookup
!
ip ssh time-out 120
ip ssh authentication-retries 3
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key Jolly-Good-Fellow address 192.168.100.254
!
crypto ipsec transform-set TS-101 esp-3des esp-sha-hmac
!
crypto map MAP-101 10 ipsec-isakmp
 set peer 192.168.100.254
 set security-association lifetime kilobytes 10000
 set security-association lifetime seconds 86000
 set transform-set TS-101
 match address AEO-101
!
no spanning-tree vlan 171
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan171
 ip address 192.168.100.253 255.255.255.0
 crypto map MAP-101
crypto engine slot 5/1
!
interface Vlan271
 no ip address
 crypto connect vlan 171
!
interface GigabitEthernet1/1
 ip address 10.80.1.254 255.255.255.0
!
interface GigabitEthernet1/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,271,1002-1005
 switchport mode trunk
 no ip address
!
interface GigabitEthernet5/1/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,171,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
interface GigabitEthernet5/1/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,271,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
```

```
ip classless
ip route 10.83.3.0 255.255.255.0 192.168.100.254
no ip http server
!
ip access-list extended AEO-101
 permit ip 10.80.0.0 0.0.255.255 10.83.0.0 0.0.255.255
!
line con 0
line vty 0 4
 login
!
end
```

Figure 23-7 Trunk Port Configuration Example



Trunk Port (Router 2) Configuration Example

The router 2 (trunk port) configuration for the configuration shown in Figure 23-7 is as follows:

```
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router-2
!
redundancy
 main-cpu
  auto-sync standard
diagnostic level complete
ip subnet-zero
!
no ip domain-lookup
!
ip ssh time-out 120
ip ssh authentication-retries 3
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key Jolly-Good-Fellow address 192.168.100.253
!
crypto ipsec transform-set TS-101 esp-3des esp-sha-hmac
!
crypto map MAP-101 10 ipsec-isakmp
 set peer 192.168.100.253
 set security-association lifetime kilobytes 10000
 set security-association lifetime seconds 86000
 set transform-set TS-101
 match address AEO-101
!
no spanning-tree vlan 171
!
interface Vlan171
 ip address 192.168.100.254 255.255.255.0
 crypto map MAP-101
 crypto engine slot 5/1
!
interface Vlan271
 no ip address
 crypto connect vlan 171
!
interface Vlan1
 no ip address
 shutdown
!
interface GigabitEthernet1/1
 ip address 10.83.3.254 255.255.255.0
!
interface GigabitEthernet1/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,271,1002-1005
 switchport mode trunk
 no ip address
!
```

```
interface GigabitEthernet5/1/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,171,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
interface GigabitEthernet5/1/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,271,1002-1005
 switchport mode trunk
 no ip address
 flowcontrol receive on
!
ip classless
ip route 10.80.1.0 255.255.255.0 192.168.100.253
no ip http server
!
ip access-list extended AEO-101
 permit ip 10.83.0.0 0.0.255.255 10.80.0.0 0.0.255.255
!
line con 0
line vty 0 4
 login
!
end
```

IPSec VPN SPA Connections to WAN Interfaces (Crypto-Connect Mode) Configuration Examples

The following are configuration examples of IPSec VPN SPA connections to WAN interfaces:

- IPSec VPN SPA Connection to an ATM Port Adapter Configuration Example, page 23-62
- IPSec VPN SPA Connection to a POS Port Adapter Configuration Example, page 23-63

IPSec VPN SPA Connection to an ATM Port Adapter Configuration Example

The following example shows the configuration of an IPSec VPN SPA connection to an ATM port adapter. In this example, note the following:

- The ATM port adapter is in slot 6, subslot 0.
- The IPSec VPN SPA is in slot 5, subslot 1.

```
interface GigabitEthernet5/1/1
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,101,1002-1005
 switchport mode trunk
 cdp enable
!
interface ATM6/0/0
 no ip address
 atm clock INTERNAL
```

```
!
interface ATM6/0/0.101 point-to-point
 pvc 1/101
 crypto connect vlan 101
!
interface Vlan101
 ip address 192.168.101.1 255.255.255.0
 no mop enabled
 crypto engine slot 5/1
!
```

IPSec VPN SPA Connection to a POS Port Adapter Configuration Example

The following example shows the configuration of an IPSec VPN SPA connection to a POS port adapter. In this example, note the following:

- The POS port adapter is in slot 6, subslot 1.
- The IPSec VPN SPA is in slot 5, subslot 1.

```
!
frame-relay switching
!
interface GigabitEthernet5/1/1
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,16,1002-1005
 switchport mode trunk
 cdp enable
!
interface POS6/1/0
 no ip address
 encapsulation frame-relay
!!!
!!! The peer POS interface config does not need
!!! to have the following two lines.
!!!
 no keepalive
 clock source internal
 frame-relay intf-type dce
!
interface POS6/1/0.16 point-to-point
 no cdp enable
 frame-relay interface-dlci 16
 crypto connect vlan 16
!
interface Vlan16
 ip address 192.168.16.1 255.255.255.0
 no mop enabled
  crypto engine slot 5/1
```

VRF Mode Using Crypto Maps Configuration Examples

The following sections provide examples of VRF mode configurations using crypto maps:

- VRF Mode Configuration Example 1 (Basic Configuration), page 23-64
- VRF Mode Configuration Example 2 (Remote Access Using Easy VPN), page 23-66
- VRF Mode Configuration Example 3 (PE), page 23-72
- VRF Mode Configuration Example 4 (CE), page 23-77
- VRF Mode Configuration Example 5 (Crypto FVRF), page 23-81

VRF Mode Configuration Example 1 (Basic Configuration)

The following example shows a basic IPSec VPN SPA configuration using crypto maps in VRF mode:

Note The IPSec VPN SPA VRF mode configuration commands are in bold.

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

```
ip vrf pepsi
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
ip vrf coke
 rd 2000:1
 route-target export 2000:1
 route-target import 2000:1
!
crypto engine mode vrf
!
crypto keyring key0
  pre-shared-key address 0.0.0.0 0.0.0.0 key happy-piggy
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp profile prof1
 vrf pepsi
 keyring key0
 match identity address 1.1.1.2 255.255.255.255
crypto isakmp profile prof2
 vrf coke
 keyring key0
 match identity address 2.2.2.2 255.255.255.255
!
crypto ipsec transform-set ts esp-3des esp-sha-hmac
 mode transport
!
crypto map map100 local-address GigabitEthernet 1/1
crypto map map100 10 ipsec-isakmp
 set peer 171.1.1.2
 set transform-set ts
 set isakmp-profile prof1
```

```
 match address 101
!
crypto map map100 local-address GigabitEthernet 1/1
crypto map map200 10 ipsec-isakmp
 set peer 171.1.1.2
 set transform-set ts
 set isakmp-profile prof2
 match address 101
!
interface GigabitEthernet3/0/1
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,100, 200, 1002-1005
 switchport mode trunk
 flowcontrol receive on
!
interface GigabitEthernet3/0/2
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 flowcontrol receive on
!
interface GigabitEthernet1/1
 ip address 171.1.1.1 255.255.255.0
 crypto engine slot 3/0 outside
!
interface GigabitEthernet2/1
 ip vrf forwarding pepsi
 ip address 10.1.1.1 255.255.255.0
!
interface GigabitEthernet2/2
 ip vrf forwarding coke
 ip address 10.1.1.1 255.255.255.0
!
.
.
.
!
interface Vlan100
 ip vrf forwarding pepsi
 ip address 10.2.1.1 255.255.255.0
 crypto engine slot 3/0 inside
 crypto map map100
!
interface Vlan200
 ip vrf forwarding coke
 ip address 10.2.1.1 255.255.255.0
 crypto engine slot 3/0 inside
 crypto map map200
!
access-list 101 permit ip 10.1.1.0 0.0.0.255 10.2.1.0 0.0.0.255
```

The following example shows that you may configure multiple outside interfaces and set the different peer addresses in the crypto maps accordingly:

```
interface GigabitEthernet1/1
 ip address 171.1.1.1 255.255.255.0
 crypto engine slot 3/0 outside
!
interface GigabitEthernet1/2
 ip address 170.1.1.1 255.255.255.0
 crypto engine slot 3/0 outside
!
crypto map map100 10 ipsec-isakmp
 set peer 171.1.1.2
 set transform-set ts
 set isakmp-profile prof1
 match address 101
!
crypto map map200 10 ipsec-isakmp
 set peer 170.1.1.2
 set transform-set ts
 set isakmp-profile prof2
 match address 101
!
```

The following example shows that you may also configure the interface VLAN to use an IP address that is different from the actual destination network subnet, and add a static route:

```
interface Vlan100
 ip vrf forwarding pepsi
 ip address 10.3.1.1 255.255.255.0
 crypto map map100
 crypto engine slot 3/0 inside
!
ip route vrf coke 10.2.1.0 255.255.255.0 Vlan 100
```

VRF Mode Configuration Example 2 (Remote Access Using Easy VPN)

The following example shows an IPSec VPN SPA configuration using VRF mode for remote access using Easy VPN:

Note The IPSec VPN SPA VRF mode configuration commands and related IPSec commands are in bold.

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service counters max age 10
!
hostname router-Alfred
!
enable password lab
!
username lab password 0 lab
```

```
username sunil@vpn1 password 0 lab
username sunil@vpn2 password 0 lab
aaa new-model
aaa authentication login vpn group radius local
aaa authorization network vpn group radius local
aaa accounting update periodic 30
aaa accounting network vpn start-stop group radius
!
aaa session-id common
ip subnet-zero
!
ip ftp username nsite
ip ftp password lab
no ip domain-lookup
!
ip vrf vpn1
 rd 100:1
 route-target export 100:1
 route-target import 100:1
!
ip vrf vpn2
 rd 101:1
 route-target export 101:1
 route-target import 101:1
!
mls ip multicast flow-stat-timer 9
no mls flow ip
no mls flow ipv6
mls qos
mls cef error action freeze
mpls label protocol ldp
mpls ldp router-id loopback0
tag-switching ip default route
!
crypto keyring vpn2
  pre-shared-key address 223.1.1.10 key vpn1123

crypto keyring vpn2
  pre-shared-key address 223.1.1.20 key vpn2123
!
crypto isakmp policy 1
 encr 3des
 authentication pre-share
 group 2
crypto isakmp keepalive 45 3

crypto isakmp profile vpn1
 vrf vpn1
 keyring vpn1
 match identity group address 223.1.1.10 255.255.255.255

crypto isakmp profile vpn1-ra
 vrf vpn1
 match identity group vpn1group
 client authentication list vpn
 isakmp authorization list vpn
 accounting vpn
crypto isakmp profile vpn2
 vrf vpn2
 keyring vpn2
 match identity address 223.1.1.20 255.255.255.255
!
crypto ipsec security-association lifetime seconds 80000
!
```

```
crypto ipsec transform-set VPN esp-3des esp-sha-hmac
!
crypto dynamic-map vpn1 1
set transform-set ipsec
set isakmp-profile vpn1-ra
!
crypto map vpn1 local-address GigabitEthernet 4/1
crypto map vpn1 10 ipsec-isakmp
 set peer 223.1.1.10
 set transform-set VPN
 set isakmp-profile vpn1
 match address ACL
 reverse-route
crypto map vpn 6000 ipsec-isakmp dynamic vpn1
crypto map vpn2 local-address GigabitEthernet4/1
crypto map vpn2 10 ipsec-isakmp
 set peer 223.1.1.20
 set transform-set VPN
 set isakmp-profile vpn2
 match address ACL
 reverse-route
!
crypto engine mode vrf
spanning-tree mode pvst
no spanning-tree optimize bpdu transmission
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
!
redundancy
 mode sso
 main-cpu
 auto-sync running-config
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
interface Loopback1
 description id for ldp
 ip address 101.1.1.1 255.255.255.252
!
interface FastEthernet3/1
 description MGMT-TFTP VLAN
 no ip address
 speed 100
 duplex full
 switchport
 switchport access vlan 700
 switchport mode access
!
....
....
!
interface GigabitEthernet4/1
 description Internet Link
 ip address 30.1.1.2 255.255.255.0
 load-interval 30
 speed nonegotiate
 crypto engine slot 5/0 outside
!
interface GigabitEthernet4/2
 no ip address
 shutdown
!
interface GigabitEthernet4/3
```

```
 no ip address
 shutdown
!
interface GigabitEthernet4/4
 description To 7200-PE-1 (g0/2)
 no ip address
 load-interval 30
 speed nonegotiate
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 201,202
 switchport mode trunk
!
interface GigabitEthernet4/5
 no ip address
 shutdown
!
interface GigabitEthernet4/6
 no ip address
 shutdown
!
interface GigabitEthernet4/7
 no ip address
 load-interval 30
 speed nonegotiate
 switchport
 switchport access vlan 701
 switchport mode access
!
interface GigabitEthernet5/0/1
 description VPNSM I-VLAN's
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,101-103,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet5/0/2
 description VPNSM P-VLAN
 no ip address
 flowcontrol receive on
 flowcontrol
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet6/1
 ip address 100.1.1.1 255.255.255.0
 mpls label protocol ldp
 tag-switching ip
!
interface GigabitEthernet6/2
 no ip address
 shutdown
!
interface Vlan101
 description Interface-VLAN for VRF vpn1
 ip vrf forwarding vpn1
 ip address 10.1.1.1 255.255.255.0
```

```
 load-interval 30
 crypto map vpn1
 crypto engine slot 5/0 inside
!
interface Vlan102
 description Interface-VLAN for VRF vpn2
 ip vrf forwarding vpn2
 ip address 10.1.1.1 255.255.255.0
 load-interval 30
 crypto map vpn2
 crypto engine slot 5/0 inside
!
interface Vlan201
 description to FWSM
 ip vrf forwarding vpn1
 ip address 11.1.1.1 255.255.255.0
!
interface Vlan202
 description TO FWSM
 ip vrf forwarding vpn2
 ip address 11.1.1.1 255.255.255.0
!
interface Vlan700
 description TFTP-MGMT Interface
 ip address 100.1.1.21 255.255.255.0
!
router ospf 1
 log-adjacency-changes
 network 101.1.1.1 0.0.0.0 area 0
 network 100.1.1.0 0.0.0.255 area 0
 !
 router bgp 1001
 bgp log-neighbor-changes
 neighbor 151.1.1.1 remote-as 1001
 neighbor 151.1.1.1 update-source Loopback0
 neighbor 153.1.1.1 remote-as 1001
 neighbor 153.1.1.1 update-source Loopback0
 neighbor 171.1.1.1 remote-as 1001
 neighbor 171.1.1.1 update-source Loopback0
 !
 address-family ipv4
 redistribute connected
 redistribute static
 no neighbor 151.1.1.1 activate
 no neighbor 153.1.1.1 activate
 no neighbor 171.1.1.1 activate
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family vpnv4
 neighbor 151.1.1.1 activate
 neighbor 151.1.1.1 send-community extended
 neighbor 153.1.1.1 activate
 neighbor 153.1.1.1 send-community both
 neighbor 171.1.1.1 activate
 neighbor 171.1.1.1 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf vpn1
 redistribute connected
 redistribute static
 redistribute eigrp 1
 no auto-summary
```

```
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf vpn2
 redistribute connected
 redistribute static
 no auto-summary
 no synchronization
 exit-address-family
!
ip local pool vpn1 20.1.0.0 20.1.5.254 group vpn1
ip local pool vpn2 20.1.0.0 20.1.5.254 group vpn2
ip classless
ip route 0.0.0.0 0.0.0.0 GigabitEthernet4/1 30.1.1.10
ip route vrf vpn1 0.0.0.0 0.0.0.0 11.1.1.2
ip route vrf vpn2 0.0.0.0 0.0.0.0 11.1.1.2
no ip http server
!
ip access-list extended ACL
 permit ip host 192.168.1.1 host 172.10.8.1
ip radius source-interface Vlan700
!
logging 100.1.1.44
access-list 10 permit any
!
snmp-server community public RO
snmp-server community nsite-ro RO
snmp-server community nsite-rw RW
snmp-server trap link ietf
snmp-server enable traps snmp authentication linkdown linkup coldstart warmstart
snmp-server enable traps chassis
snmp-server enable traps module
snmp-server enable traps tty
snmp-server enable traps casa
snmp-server enable traps vtp
snmp-server enable traps vlancreate
snmp-server enable traps vlandelete
snmp-server enable traps bgp
snmp-server enable traps syslog
snmp-server enable traps rtr
snmp-server enable traps isakmp policy add
snmp-server enable traps isakmp policy delete
snmp-server enable traps isakmp tunnel start
snmp-server enable traps isakmp tunnel stop
snmp-server enable traps ipsec cryptomap add
snmp-server enable traps ipsec cryptomap delete
snmp-server enable traps ipsec cryptomap attach
snmp-server enable traps ipsec cryptomap detach
snmp-server enable traps ipsec tunnel start
snmp-server enable traps ipsec tunnel stop
snmp-server enable traps ipsec too-many-sas
snmp-server enable traps srp
snmp-server enable traps sonet
snmp-server enable traps mpls traffic-eng
snmp-server enable traps mpls ldp
snmp-server enable traps voice poor-qov
snmp-server enable traps mpls vpn
snmp-server host 100.1.1.1 version 2c public
snmp-server host 100.1.1.44 public
snmp-server host 100.1.1.44 version 2c trap
!
radius-server host 100.1.1.4 auth-port 1645 acct-port 1646
radius-server source-ports 1645-1646
radius-server key cisco
```

```
radius-server vsa send accounting
!
dial-peer cor custom
!
line con 0
 exec-timeout 0 0
line vty 0 4
 exec-timeout 0 0
 password lab
```

VRF Mode Configuration Example 3 (PE)

The following example shows an IPSec VPN SPA VRF mode configuration for a PE:

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

```
version 12.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
service counters max age 10
!
hostname router-Alice
!
2
logging snmp-authfail
enable password cisco
!
no aaa new-model
clock timezone pst -7
ip subnet-zero
!
no ip domain-lookup
!
ip vrf blue
 rd 300:10
 route-target export 300:10
 route-target import 300:10
!
ip vrf red
 rd 100:10
 route-target export 200:10
 route-target import 200:10
!
ip multicast-routing
ip multicast-routing vrf red
mpls label protocol ldp
mls ip multicast flow-stat-timer 9
no mls flow ip
no mls flow ipv6
mls cef error action freeze
!
crypto keyring test
  pre-shared-key address 10.1.1.2 key cisco
!
crypto isakmp policy 10
 encr 3des
 authentication pre-share
```

```
crypto isakmp key cisco address 192.168.32.2
crypto isakmp key cisco address 11.1.1.2
crypto isakmp key cisco address 192.168.31.2
crypto isakmp keepalive 10
!
crypto ipsec transform-set test esp-3des esp-md5-hmac
crypto ipsec transform-set repro esp-3des esp-sha-hmac
!
crypto ipsec profile red
 set transform-set test
!
crypto ipsec profile test
 set transform-set test
!
crypto map local-address Pos3/1
crypto map test 10 ipsec-isakmp
 ! Incomplete
 set peer 10.1.1.2
 set transform-set test
 match address 101
!
crypto map local-address Pos3/1
crypto map repro 10 ipsec-isakmp
 set peer 192.168.32.2
 set transform-set repro
 match address repro
!
crypto engine mode vrf
!
power redundancy-mode combined
spanning-tree mode pvst
no spanning-tree optimize bpdu transmission
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
  auto-sync standard
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
interface Loopback0
 ip address 192.168.1.1 255.255.255.255
!interface GigabitEthernet2/
  ip address 192.168.31.155.255.255.0
 crypto engine slot 4/0

interface GigabitEthernet2/2
 no ip address
 shutdown
!
.
.
.
!
interface GigabitEthernet2/16
 no ip address
 shutdown
!
interface GigabitEthernet3/1
 no ip address
```

```
 shutdown
!
interface GigabitEthernet3/2
 no ip address
 shutdown
!
interface GigabitEthernet3/3
 no ip address
 shutdown
!
interface GigabitEthernet3/4
 no ip address
 shutdown
!
interface POS3/1
 ip address 192.168.32.1 255.255.255.0
 mls qos trust dscp
 clock source internal
 crypto engine slot 4/0 outside
!
interface POS3/2
 no ip address
 shutdown
 mls qos trust dscp
!
interface POS3/3
 no ip address
 shutdown
 mls qos trust dscp
!
interface POS3/4
 no ip address
 shutdown
 mls qos trust dscp
!
interface GigabitEthernet4/0/1
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet4/0/2
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet5/1
 no ip address
 shutdown
!
interface GigabitEthernet5/2
 no ip address
 shutdown
!
interface GigabitEthernet7/1
```

```
 ip address 17.8.15.1 255.255.0.0
!
interface GigabitEthernet7/2
 no ip address
 shutdown
!
interface GigabitEthernet7/3
 no ip address
 shutdown
.
.
.
!
interface GigabitEthernet7/9
 no ip address
 shutdown
!
interface GigabitEthernet7/10
 ip address 10.1.1.1 255.255.255.0
 crypto engine slot 4/0
!
interface GigabitEthernet7/11
 ip address 11.1.1.1 255.255.255.0
 crypto engine slot 4/0
!
interface GigabitEthernet7/12
 no ip address
 shutdown
.
.
.
!
interface GigabitEthernet7/19
 no ip address
 shutdown
!
interface GigabitEthernet7/20
 ip address 192.168.30.1 255.255.255.0
 mpls label protocol ldp
 tag-switching ip
!
interface GigabitEthernet7/21
 no ip address
 shutdown
.
.
.
!
interface GigabitEthernet7/41
 ip vrf forwarding red
 ip address 192.168.41.1 255.255.255.0
 ip pim sparse-dense-mode
!
interface GigabitEthernet7/42
 no ip address
.
.
.
!
interface GigabitEthernet7/48
 no ip address
 shutdown
!
interface Vlan1
```

```
 no ip address
 shutdown
!
interface Vlan32
 no ip address
 no mop enabled
 crypto map repro
!
interface Vlan100
 no ip address
 no mop enabled
!
interface Vlan110
 no ip address
 no mop enabled
!
router ospf 1
 log-adjacency-changes
 passive-interface GigabitEthernet7/10
 network 10.0.0.0 0.255.255.255 area 0
 network 192.168.1.0 0.0.0.255 area 0
 network 192.168.30.0 0.0.0.255 area 0
!
router ospf 10 vrf red
 log-adjacency-changes
 redistribute bgp 1 subnets
 redistribute rip subnets
 network 10.2.1.0 0.0.0.255 area 0
!
router rip
 version 2
!
 address-family ipv4 vrf red
 redistribute ospf 10 metric 10
 redistribute bgp 1 metric 10
 network 31.0.0.0
 network 32.0.0.0
 network 192.168.41.0
 no auto-summary
 exit-address-family
!
router bgp 1
 no synchronization
 bgp log-neighbor-changes
 neighbor 192.168.3.1 remote-as 1
 neighbor 192.168.3.1 update-source Loopback0
 no auto-summary
!
 address-family vpnv4
 neighbor 192.168.3.1 activate
 neighbor 192.168.3.1 send-community extended
 exit-address-family
!
 address-family ipv4 vrf red
 redistribute ospf 10
 redistribute rip
 no auto-summary
 no synchronization
 exit-address-family
!
 address-family ipv4 vrf blue
 neighbor 11.2.1.2 remote-as 65001
 neighbor 11.2.1.2 activate
 no auto-summary
```

```
 no synchronization
 network 11.2.1.0 mask 255.255.255.0
 exit-address-family
!
ip classless
ip route 0.0.0.0 0.0.0.0 17.8.0.1
ip route 192.168.9.0 255.255.255.0 Tunnel32
ip route 192.168.43.0 255.255.255.0 Tunnel32
no ip http server
!
ip access-list extended repro
 permit gre host 192.168.32.1 host 192.168.32.2
ip access-list extended to2651
ip access-list extended to3745
ip access-list extended to7609
!
access-list 199 permit ip host 10.1.1.2 host 192.168.6.1
!
control-plane
!
dial-peer cor custom
!
line con 0
 exec-timeout 0 0
line vty 0 4
 login
!
end
```

VRF Mode Configuration Example 4 (CE)

The following example shows an IPSec VPN SPA VRF mode configuration for a CE:

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service counters max age 10
!
hostname router-Bobby
!
enable password cisco
!
no aaa new-model
clock timezone pst -7
ip subnet-zero
!
ip multicast-routing
!
crypto isakmp policy 10
 encr 3des
 authentication pre-share
crypto isakmp key cisco address 192.168.32.1
crypto isakmp key cisco address 192.168.31.1
!
crypto ipsec transform-set repro esp-3des esp-md5-hmac
```

```
crypto ipsec transform-set test esp-3des esp-md5-hmac
!
crypto ipsec profile test
 set transform-set test
!
crypto map repro 10 ipsec-isakmp
 set peer 192.168.32.1
 set transform-set repro
 match address repro
!
crypto map test 10 ipsec-isakmp
 set peer 192.168.31.1
 set transform-set test
 match address tope
!
spanning-tree mode pvst
spanning-tree extend system-id
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
!
redundancy
 mode sso
 main-cpu
  auto-sync running-config
!
vlan internal allocation policy ascending
!
interface Loopback0
 ip address 192.168.9.1 255.255.255.0
!
interface GigabitEthernet1/1
 no ip address
 shutdown
!
interface GigabitEthernet1/2
 no ip address
 shutdown
!
interface GigabitEthernet2/1
 no ip address
 crypto connect vlan 31
!
interface GigabitEthernet2/2
 no ip address
 shutdown
.
.
.
!
interface GigabitEthernet2/16
 no ip address
 shutdown
!
interface GigabitEthernet3/1
 no ip address
 shutdown
 flowcontrol receive on
 flowcontrol send off
!
interface GigabitEthernet3/2
 no ip address
 shutdown
 flowcontrol receive on
 flowcontrol send off
```

```
!
interface GigabitEthernet3/3
 no ip address
 shutdown
!
interface GigabitEthernet3/4
 no ip address
 shutdown
!
interface POS3/1
 no ip address
 mls qos trust dscp
 crypto connect vlan 32
!
interface POS3/2
 no ip address
 shutdown
 mls qos trust dscp
!
interface POS3/3
 no ip address
 shutdown
 mls qos trust dscp
!
interface POS3/4
 no ip address
 shutdown
 mls qos trust dscp
!
interface GigabitEthernet4/0/1
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,31,32,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet4/0/2
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet7/1
 ip address 17.8.15.9 255.255.0.0
!
interface GigabitEthernet7/2
 no ip address
 shutdown
.
.
.
!
interface GigabitEthernet7/42
 no ip address
 shutdown
!
interface GigabitEthernet7/43
```

```
 ip address 192.168.43.1 255.255.255.0
 ip pim sparse-dense-mode
!
interface GigabitEthernet7/44
 no ip address
 shutdown
!
interface GigabitEthernet7/45
 no ip address
 shutdown
!
interface GigabitEthernet7/46
 no ip address
 shutdown
!
interface GigabitEthernet7/47
 no ip address
 shutdown
!
interface GigabitEthernet7/48
 no ip address
 shutdown
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan31
 ip address 192.168.31.2 255.255.255.0
 no mop enabled
 crypto map test
 crypto engine slot 4/0
!
interface Vlan32
 ip address 192.168.32.2 255.255.255.0
 no mop enabled
 crypto map repro
 crypto engine slot 4/0
!
router rip
 version 2
 network 31.0.0.0
 network 32.0.0.0
 network 192.168.9.0
 network 192.168.43.0
 distribute-list 1 out
 no auto-summary
!
ip classless
ip route 192.168.6.0 255.255.255.0 Tunnel32
no ip http server
!
ip access-list extended repro
 permit gre host 192.168.32.2 host 192.168.32.1
ip access-list extended tope
 permit gre host 192.168.31.2 host 192.168.31.1
!
access-list 1 permit 192.168.9.0 0.0.0.255
access-list 1 permit 192.168.43.0 0.0.0.255
!
dial-peer cor custom
!
line con 0
 exec-timeout 0 0
```

```
line vty 0 4
 no login
 transport input lat pad mop telnet rlogin udptn nasi
!
!
end
```

VRF Mode Configuration Example 5 (Crypto FVRF)

The following example shows a sample configuration with crypto FVRF support, including two VRFs and one separate FVRF. This configuration has the following three VRFs:

- VRF coke, which is purely an inside VRF and uses coke-fvrf as its outside FVRF
- VRF coke-fvrf, which is purely a front door VRF
- VRF pepsi, which is both an inside and outside VRF

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

```
ip vrf pepsi
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
ip vrf coke
 rd 2000:1
 route-target export 2000:1
 route-target import 2000:1
!
ip vrf coke-fvrf
 rd 3000:1
 route-target export 3000:1
 route-target export 3000:1
!
crypto engine mode vrf
!
crypto keyring pepsikey vrf pepsi
  pre-shared-key address 0.0.0.0 0.0.0.0 key happy-ramki
!
crypto keyring cokekey vrf coke-fvrf
  pre-shared-key address 0.0.0.0 0.0.0.0 key happy-mo
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp profile prof1
   vrf pepsi
   keyring pepsikey
   match identity address 1.1.1.2 255.255.255.255 pepsi
crypto isakmp profile prof2
   vrf coke
   keyring cokekey
   match identity address 2.2.2.2 255.255.255.255 coke-fvrf
!
crypto ipsec transform-set ts esp-3des esp-sha-hmac
 mode transport
!
crypto map map local-address GigabitEthernet1/1
crypto map map100 10 ipsec-isakmp
```

```
 set peer 171.1.1.2
set transform-set ts
set isakmp-profile prof1
 match address 101
!
crypto map map local-address GigabitEthernet1/2
crypto map map200 10 ipsec-isakmp
 set peer 171.1.1.2
set transform-set ts
set isakmp-profile prof2
 match address 101
!
interface GigabitEthernet3/1
 description VPNSM interface HMAC
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,100, 200, 1002-1005
 switchport mode trunk
 flowcontrol receive on
!
interface GigabitEthernet3/2
 mtu 4500
 no ip address
 snmp trap link-status
switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 flowcontrol receive on
!
interface GigabitEthernet1/1
   description Front door WAN interface for pepsi to the internet
   ip address 171.1.1.1 255.255.255.0
   ip vrf forwarding pepsi
   crypto engine slot 3 outside
!
interface GigabitEthernet1/2
   description Front door WAN interface for coke to the internet
   ip address 171.1.1.1 255.255.255.0
   ip vrf forwarding coke-fvrf
   crypto engine slot 3 outside
!
interface GigabitEthernet2/1
   description Inside LAN interface for pepsi
   ip address 10.1.1.1 255.255.255.0
   ip vrf forwarding pepsi
!
interface GigabitEthernet2/2
   description Inside LAN interface for coke
   ip address 10.1.1.1 255.255.255.0
   ip vrf forwarding coke
!
interface Vlan100
   description Interface vlan for pepsi
   ip address 10.2.1.1 255.255.255.0
   ip vrf forwarding pepsi
   crypto engine slot 3 inside
   crypto map map100
!
interface Vlan200
   description Interface vlan for coke
```

```
   ip address 10.2.1.1 255.255.255.0
   ip vrf forwarding coke
   crypto engine slot 3 inside
   crypto map map200
!
access-list 101 permit ip 10.1.1.0 0.0.0.255 10.2.1.0 0.0.0.255
```

Users may also configure the interface VLAN to use an IP address that is different from the actual destination network subnet, and add a static route.

```
!
interface Vlan100
   ip address 10.3.1.1 255.255.255.0
   ip vrf forwarding pepsi
   crypto map map100
   crypto engine slot 3
!
ip route vrf pepsi 10.2.1.0 255.255.255.0 Vlan 100
!
```

GRE Tunneling in Crypto-Connect Mode Configuration Examples

This section provides examples of GRE tunneling configurations in crypto-connect mode:

- GRE Tunneling (Router 1) Configuration Example, page 23-83
- GRE Tunneling (Router 2) Configuration Example, page 23-84

In both routers, the IPSec VPN SPA is in slot 5, subslot 1. Gigabit Ethernet interfaces 5/0/1 and 5/0/2 are the secured ports, and Gigabit Ethernet interfaces 1/1 and 1/2 are the LAN ports.

GRE Tunneling (Router 1) Configuration Example

The following example shows the configuration of GRE tunneling for router 1:

```
crypto isakmp policy 100
 encr 3des
 authentication pre-share
crypto isakmp key 12345 address 192.168.1.0 255.255.255.0
!
crypto ipsec transform-set ts esp-3des esp-sha-hmac
!
crypto map cm1 100 ipsec-isakmp
 set peer 192.168.1.1
 set security-association level per-host
 set security-association lifetime kilobytes 536870912
 set security-association lifetime seconds 86400
 set transform-set ts
 match address acl1
!
interface GigabitEthernet5/0/2
 no ip address
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,502,1002-1005
 switchport mode trunk
!
interface GigabitEthernet1/2
 ip address 5.0.0.254 255.255.255.0
!
interface GigabitEthernet5/0/1
```

```
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2,1002-1005
 switchport mode trunk
 cdp enable
!
interface Vlan2
 ip address 192.168.1.254 255.255.255.0
 no mop enabled
 crypto map cm1
 crypto engine slot 5/0
!
interface Vlan502
 no ip address
 crypto connect vlan 2
!
interface Tunnel1
 ip address 10.1.1.254 255.255.255.0
 tunnel source vlan2
 tunnel destination 192.168.1.1
!
ip route 6.0.0.0 255.255.255.0 Tunnel1
!
ip access-list extended acl1
 permit gre host 192.168.1.254 host 192.168.1.1
!
```

GRE Tunneling (Router 2) Configuration Example

The following example shows the configuration of GRE tunneling for router 2:

```
crypto isakmp policy 100
 encr 3des
 authentication pre-share
crypto isakmp key 12345 address 192.168.1.0 255.255.255.0
!
crypto ipsec transform-set ts esp-3des esp-sha-hmac
!
crypto map cm1 100 ipsec-isakmp
 set peer 192.168.1.254
 set security-association level per-host
 set security-association lifetime kilobytes 536870912
 set security-association lifetime seconds 86400
 set transform-set ts
 match address acl1
!
interface GigabitEthernet5/0/2
 no ip address
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,502,1002-1005
 switchport mode trunk
!
interface GigabitEthernet1/2
 ip address 6.0.0.254 255.255.255.0
!
interface GigabitEthernet5/0/1
 no ip address
 flowcontrol receive on
 flowcontrol send off
```

```
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2,1002-1005
 switchport mode trunk
 cdp enable
!
interface Vlan2
 ip address 192.168.1.1 255.255.255.0
 no mop enabled
 crypto map cm1
  crypto engine slot 5/0
!
interface Vlan502
 no ip address
 crypto connect vlan 2
!
interface Tunnel1
 ip address 10.1.1.1 255.255.255.0
 tunnel source vlan2
 tunnel destination 192.168.1.254
!
ip route 5.0.0.0 255.255.255.0 Tunnel1
!
ip access-list extended acl1
 permit gre host 192.168.1.1 host 192.168.1.254
!
```

VRF Mode with Tunnel Protection Configuration Example

The following examples show VRF mode configuration that uses tunnel protection:

- VRF Mode with Tunnel Protection Configuration Example 1, page 23-85
- VRF Mode with Tunnel Protection Configuration Example 2, page 23-87

VRF Mode with Tunnel Protection Configuration Example 1

The following example shows a VRF mode configuration that uses tunnel protection:

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

```
ip vrf coke
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
crypto keyring key1
 pre-shared-key address 100.1.1.1 key happy-eddie
!
crypto isakmp policy 1
 authentication pre-share
 lifetime 500

crypto isakmp profile prof1
 vrf coke
 keyring key1
```

```
 match identity address 100.1.1.1 255.255.255.255
!
crypto ipsec transform-set TR esp-des esp-md5-hmac
!
crypto ipsec profile tp
 set transform-set TR
 set isakmp-profile prof1
!
crypto map M10k local-address GigabitEthernet6/1

crypto map M10k 1 ipsec-isakmp
 set peer 100.1.1.1
  set transform-set TR
 set isakmp-profile prof1
 match address 110
!
crypto engine mode vrf
!
interface Tunnel1
 ip vrf forwarding coke
 ip address 10.1.1.254 255.255.255.0
 tunnel source 172.1.1.1
 tunnel destination 100.1.1.1
 tunnel protection ipsec profile tp
 crypto engine slot 4/0 inside
!
interface GigabitEthernet4/0/1
  flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 cdp enable
 spanning-tree portfast trunk
!
interface GigabitEthernet4/0/2
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 cdp enable
 spanning-tree portfast trunk
!
interface GigabitEthernet6/1
 ip address 172.1.1.1 255.255.255.0
 crypto engine slot 4/0 outside
!
interface FastEthernet7/13
 ip vrf forwarding coke
 ip address 13.1.1.2 255.255.255.0
!
ip route 100.1.1.1 255.255.255.255 172.1.1.254

access-list 110 permit ip host 13.1.1.254 host 10.1.1.2
```

VRF Mode with Tunnel Protection Configuration Example 2

The following example shows a VRF mode configuration that uses tunnel protection:

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

```
! iVrf:  coke
! fVrf:  coke-fvrf
!
ip vrf coke-fvrf
 rd 3000:1
 route-target export 3000:1
 route-target export 3000:1

ip vrf coke
 rd 2000:1
 route-target export 2000:1
 route-target import 2000:1
!
crypto engine mode vrf
!
crypto keyring cokekey vrf coke-fvrf
  pre-shared-key address 0.0.0.0 0.0.0.0 key happy-ramki
!
crypto isakmp policy 1
 authentication pre-share
!
crypto isakmp profile prof2
   vrf coke
   keyring cokekey
   match identity address 10.0.0.2 255.255.255.255 coke-fvrf
!
crypto ipsec transform-set ts esp-3des esp-sha-hmac
 mode transport
!
interface GigabitEthernet1/2
   description Front door WAN interface for coke to the internet
   ip address 10.0.0.1 255.255.255.0
   ip vrf forwarding coke-fvrf
   crypto engine slot 3/0 outside
!
interface GigabitEthernet2/2
   description Inside LAN interface for coke
   ip address 10.1.1.1 255.255.255.0
   ip vrf forwarding coke
!
Interface Tunnel102
   ip vrf forwarding coke
   ip address 32.0.0.1 255.255.255.0
   tunnel source 10.0.0.1
   tunnel destination 10.0.0.2
   tunnel vrf pepsi
   tunnel protection ipsec profile ipsecprof_coke
   crypto engine slot 3/0 inside
!
ip route vrf coke 40.0.0.0 255.255.255.0 Tunnel102
```

GRE Takeover Criteria Configuration Examples

The following examples show how to configure the GRE takeover criteria:

- GRE Takeover Criteria Global Configuration Example, page 23-88
- GRE Takeover Criteria Tunnel Configuration Example, page 23-88

GRE Takeover Criteria Global Configuration Example

The following example shows that the GRE takeover criteria has been set globally and the Supervisor Engine hardware or RP always does the GRE processing:

```
Router(config)# crypto engine gre supervisor
```

GRE Takeover Criteria Tunnel Configuration Example

The following example shows that the GRE takeover criteria has been set individually for tunnel interface 3 and the IPSec VPN SPA always does the GRE processing for this tunnel:

```
Router(config)# interface tunnel 3
Router(config-if)# crypto engine gre vpnblade
```

IP Multicast Over a GRE Tunnel Configuration Example

The following example shows how to configure single-SPA mode for multicast traffic:

```
Router# hw-module slot 1 subslot 1 only
```

The following example shows a back-to-back configuration that is configured for IP multicast over GRE:

```
=========uut1 config ======================================================
ip multicast-routing
mls ip multicast
!
crypto isakmp policy 5
encr 3des
authentication pre-share
group 2
lifetime 14400
!
crypto isakmp key 12345 address 0.0.0.0 0.0.0.0
crypto ipsec transform-set t3 esp-3des esp-sha-hmac
!
crypto map map_t3 10 ipsec-isakmp
set peer 2.20.120.10
set transform-set t3
match address acl_t3
!
interface Loopback3
ip address 1.120.1.3 255.255.255.255
!
interface Tunnel15
ip address 15.15.1.1 255.255.255.0
ip mtu 9216
ip pim sparse-dense-mode
tunnel source Loopback3
tunnel destination 1.120.22.1
```

```
!
interface GigabitEthernet2/1/1
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 1,100,1002-1005
switchport mode trunk
mtu 9216
no ip address
flowcontrol receive on
flowcontrol send off
no cdp enable
spanning-tree portfast trunk
!
interface GigabitEthernet2/1/2
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 1,1002-1005
switchport mode trunk
mtu 9216
no ip address
flowcontrol receive on
flowcontrol send off
no cdp enable
spanning-tree portfast trunk
!
interface GigabitEthernet8/1
crypto connect vlan 100
!
interface GigabitEthernet8/2
ip address 120.1.0.5 255.255.255.0
!
interface Vlan100
ip address 120.0.0.1 255.255.255.0
logging event link-status
no mop enabled
crypto map map_t3
crypto engine slot 2/1
!
router ospf 10
log-adjacency-changes
network 15.15.1.0 0.0.0.255 area 0
network 120.0.0.0 0.0.0.255 area 0

======uut2 config====
ip multicast-routing
mls ip multicast flow-stat-timer 9
!
crypto isakmp policy 5
encr 3des
authentication pre-share
group 2
lifetime 14400
!
crypto isakmp key 12345 address 0.0.0.0 0.0.0.0
crypto ipsec transform-set t3 esp-3des esp-sha-hmac
!
crypto map map_t3 10 ipsec-isakmp
set peer 120.0.0.1
set transform-set t3
match address acl_t3
!
interface Loopback0
ip address 1.120.22.1 255.255.255.255
!
```

```
interface Tunnel15
ip address 15.15.1.2 255.255.255.0
ip mtu 9216
ip pim sparse-dense-mode
tunnel source Loopback0
tunnel destination 1.120.1.3
!
interface GigabitEthernet3/1
crypto connect vlan 100
interface GigabitEthernet3/2
ip address 121.0.0.1 255.255.255.0
ip pim sparse-dense-mode
ip igmp version 3
logging event link-status
!
interface GigabitEthernet7/0/1
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 1,100,1002-1005
switchport mode trunk
mtu 9216
no ip address
mls qos trust cos
flowcontrol receive on
flowcontrol send off
spanning-tree portfast trunk
!
interface GigabitEthernet7/0/2
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 1,1002-1005
switchport mode trunk
mtu 9216
no ip address
mls qos trust cos
flowcontrol receive on
flowcontrol send off
spanning-tree portfast trunk
!
interface Vlan100
ip address 120.0.0.2 255.255.255.0
logging event link-status
crypto map map_t3
crypto engine slot 7/0
!
router ospf 10
log-adjacency-changes
network 15.15.1.0 0.0.0.255 area 0
network 121.0.0.0 0.0.0.255 area 0
!
ip pim send-rp-discovery Tunnel15 scope 10
```

IPSec Virtual Tunnel Interfaces Configuration Example

The following examples show VRF mode configurations that use VTI:

- IPSec Virtual Tunnel Interface Configuration Example 1, page 23-91
- IPSec Virtual Tunnel Interface Configuration Example 2 (FVRF), page 23-93
- IPSec Virtual Tunnel Interface Configuration Example 3 (CCA), page 23-97

IPSec Virtual Tunnel Interface Configuration Example 1

The following example configuration uses a preshared key for authentication between peers. VPN traffic is forwarded to the IPSec virtual tunnel interface for encryption and then sent out of the physical interface. The tunnel checks packets for IPSec policy and passes them to the IPSec VPN SPA for IPSec encapsulation.

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

Router 1 Configuration

```
service timestamps debug datetime
service timestamps log datetime
hostname 7600-3
no aaa new-model
ip subnet-zero
ip cef
controller ISA 6/1
!
crypto isakmp policy 1
encr 3des
authentication pre-share
group 2
crypto isakmp key Cisco12345 address 0.0.0.0 0.0.0.0
crypto ipsec transform-set T1 esp-3des esp-sha-hmac
crypto ipsec profile P1
set transform-set T1
!
interface Tunnel0
ip address 10.0.51.203 255.255.255.0
ip ospf mtu-ignore
load-interval 30
tunnel source 10.0.149.203
tunnel destination 10.0.149.217
tunnel mode ipsec ipv4
tunnel protection ipsec profile P1
crypto engine slot 4/0 inside
!
interface Ethernet3/0
ip address 10.0.149.203 255.255.255.0
duplex full
crypto engine slot 4/0 outside
!
interface Ethernet3/3
ip address 10.0.35.203 255.255.255.0
duplex full
!
ip classless
ip route 10.0.36.0 255.255.255.0 Tunnel0
line con 0
line aux 0
line vty 0 4
end
```

Router 2 Configuration

```
hostname c7600-17
no aaa new-model
ip subnet-zero
ip cef
crypto isakmp policy 1
encr 3des
authentication pre-share
group 2
crypto isakmp key Cisco12345 address 0.0.0.0 0.0.0.0
crypto ipsec transform-set T1 esp-3des esp-sha-hmac
crypto ipsec profile P1
set transform-set T1
!
interface Tunnel0
ip address 10.0.51.217 255.255.255.0
ip ospf mtu-ignore
tunnel source 10.0.149.217
tunnel destination 10.0.149.203
tunnel mode ipsec ipv4
tunnel protection ipsec profile P1
crypto engine slot 4/0 inside
!
interface FastEthernet0/0
ip address 10.0.149.217 255.255.255.0
speed 100
full-duplex
crypto engine slot 4/0 outside
!
interface Ethernet1/0
ip address 10.0.36.217 255.255.255.0
load-interval 30
full-duplex
ip classless
ip route 10.0.35.0 255.255.255.0 Tunnel0
line con 0
line aux 0
line vty 0 4
end
```

IPSec Virtual Tunnel Interface Configuration Example 2 (FVRF)

The following example configuration shows an FVRF VTI configuration:

Note MPLS tunnel recirculation must be enabled when a Cisco 7600 SIP-600 is installed and VRF is to be enabled. You must add the **mls mpls tunnel-recir** command before entering the **crypto engine mode vrf** command if a Cisco 7600 SIP-600 is present in the chassis.

Router 1 Configuration

```
upgrade fpd auto
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service counters max age 10
!
hostname UPNISAD1
!
boot-start-marker
boot system disk0:
boot-end-marker
!
logging buffered 100000000 debugging
no logging console
!
no aaa new-model
ip subnet-zero
!
ip vrf fvrf1
rd 2000:1
 route-target export 2000:1
 route-target import 2000:1
!
ip vrf vrf1
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
vtp domain none
vtp mode transparent
mls ip multicast flow-stat-timer 9
no mls flow ip
no mls flow ipv6
no mls acl tcam share-global
mls cef error action freeze
crypto engine mode vrf
!
redundancy

keepalive-enable
 mode sso
 main-cpu

 auto-sync running-config
spanning-tree mode pvst
!
power redundancy-mode combined
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
fabric buffer-reserve queue
```

```
port-channel per-module load-balance
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
vlan 1
tb-vlan1 1002
tb-vlan2 1003
!
vlan 1001
!
vlan 1002
tb-vlan1 1
tb-vlan2 1003
!
vlan 1003
tb-vlan1 1
 tb-vlan2 1002
 parent 1005
!
vlan 1004
bridge 1
stp type ibm
!
vlan 1005
bridge 1
!
crypto keyring fvrf-key vrf fvrf1
pre-shared-key address 2.0.0.1 key 12345
!
crypto isakmp policy 1
encr 3des
 hash md5
 authentication pre-share
 lifetime 3600
crypto isakmp profile fvrf-prof1
   vrf vrf1
   keyring fvrf-key
   match identity address 2.0.0.1 255.255.255.255 fvrf1
!
crypto ipsec transform-set ts esp-3des
!
crypto ipsec profile fvrf-prof1
set transform-set ts
set isakmp-profile fvrf-prof1
!
interface Tunnel1
ip vrf forwarding vrf1
 ip address 10.0.0.1 255.255.255.0
 tunnel source 4.0.0.1
 tunnel destination 2.0.0.1
 tunnel mode ipsec ipv4
 tunnel vrf fvrf1
 tunnel protection ipsec profile fvrf-prof1
 crypto engine slot 9/0 inside
!
interface Loopback1
 ip vrf forwarding fvrf1
 ip address 4.0.0.1 255.255.255.0
!
interface Vlan1001
 ip vrf forwarding fvrf1
 ip address 1.0.0.1 255.255.255.0
 crypto engine slot 9/0 outside
```

```
!
ip classless
ip route vrf fvrf1 2.0.0.0 255.255.255.0 1.0.0.2
!
control-plane
!
dial-peer cor custom
!
line con 0
exec-timeout 0 0
line vty 0 4
login
!
end
```

Router 2 Configuration

```
upgrade fpd auto
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service internal
service counters max age 10
!
hostname UPNISAD2
!
boot-start-marker
boot system disk1:
boot-end-marker
!
logging buffered 100000000 debugging
no logging console
!
no aaa new-model
ip subnet-zero
!
vtp domain none
vtp mode transparent
mls ip multicast flow-stat-timer 9
no mls flow ip
no mls flow ipv6
no mls acl tcam share-global
mls cef error action freeze
crypto engine mode vrf
!
redundancy
keepalive-enable
 mode sso
 main-cpu
auto-sync running-config
spanning-tree mode pvst
spanning-tree extend system-id
!
power redundancy-mode combined
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
fabric buffer-reserve queue
port-channel per-module load-balance
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
!
vlan 1
```

```
tb-vlan1 1002
tb-vlan2 1003
!
vlan 2-1001
!
vlan 1002
tb-vlan1 1
tb-vlan2 1003
!
vlan 1003
tb-vlan1 1
 tb-vlan2 1002
 parent 1005
!
vlan 1004
 bridge 1
 stp type ibm
!
vlan 1005
bridge 1
!
vlan 3048-3072
!
crypto isakmp policy 1
encr 3des
 hash md5
 authentication pre-share
 lifetime 3600
crypto isakmp key 12345 address 4.0.0.1
!
crypto ipsec transform-set ts esp-3des
!
crypto ipsec profile vpnprof
set transform-set ts
!
interface Tunnel1
ip address 10.0.0.2 255.255.255.0
 tunnel source 2.0.0.1
 tunnel destination 4.0.0.1
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile vpnprof
 crypto engine slot 7/0 inside
!
interface GigabitEthernet6/36
switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 2-1000,3048-3072
 switchport mode trunk
!
interface Vlan2
 ip address 2.0.0.1 255.255.255.0
 crypto engine slot 7/0 outside
!
ip route 4.0.0.0 255.255.255.0 2.0.0.2
!
control-plane
!
dial-peer cor custom
!
```

```
line con 0
exec-timeout 0 0
line vty 0 4
login
!
scheduler runtime netinput 300
end
```

IPSec Virtual Tunnel Interface Configuration Example 3 (CCA)

The following example configuration shows IPSec VTI configuration using crypto connect alternative (CCA) mode:

Router 1 Configuration

```
!
crypto engine mode vrf
!
crypto keyring key1
  pre-shared-key address 14.0.0.2 key 12345
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share

!
crypto isakmp profile prof1
   keyring key1
   match identity address 14.0.0.2 255.255.255.255
!
crypto ipsec transform-set t-set1 esp-3des esp-sha-hmac
!
crypto ipsec profile prof1
 set transform-set t-set1
 set isakmp-profile prof1
!
!
interface Tunnel1
 ip address 122.0.0.2 255.255.255.0
 tunnel source 15.0.0.2
 tunnel destination 14.0.0.2
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile prof1
 crypto engine slot 2/0 inside
!
interface Loopback2
 ip address 15.0.0.2 255.255.255.0
!

interface GigabitEthernet1/3
 ip address 172.2.1.1 255.255.255.0
 crypto engine slot 2/0 outside
!
interface GigabitEthernet2/0/1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
```

```
 flowcontrol send off
 spanning-tree portfast trunk
!
interface GigabitEthernet2/0/2
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 mtu 9216
 flowcontrol receive on
 flowcontrol send off
 spanning-tree portfast trunk
!
!
ip route 14.0.0.0 255.0.0.0 172.2.1.2
ip route 172.0.0.0 255.0.0.0 172.2.1.2
```

CHAPTER 24

Configuring IKE Features Using the IPSec VPN SPA

This chapter provides information about configuring Internet Key Exchange (IKE) related features using the IPSec VPN SPA on the Cisco 7600 series router. It includes the following sections:

Note For detailed information on Internet Key Exchange (IKE), refer to the *Cisco IOS Security Configuration Guide* and *Cisco IOS Security Command Reference*.

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Tip To ensure a successful configuration of your VPN using the IPSec VPN SPA, read all of the configuration summaries and guidelines before you perform any configuration tasks.

Overview of IKE

Internet Key Exchange (IKE) is a key management protocol standard that is used in conjunction with the IPSec standard. IPSec can be configured without IKE, but IKE enhances IPSec by providing additional features, flexibility, and ease of configuration for the IPSec standard.

IKE automatically negotiates IPSec security associations (SAs) and enables IPSec secure communications without costly manual preconfiguration. Specifically, IKE provides the following benefits:

- Eliminates the need to manually specify all the IPSec security parameters in the crypto maps at both peers.
- Allows you to specify a lifetime for the IPSec SA.
- Allows encryption keys to change during IPSec sessions.
- Allows IPSec to provide anti-replay services.
- Permits certification authority (CA) support for a manageable, scalable IPSec implementation.
- Allows dynamic authentication of peers.

Because IKE negotiations must be protected, each IKE negotiation begins by agreement of both peers on a common (shared) IKE policy. This policy states which security parameters will be used to protect subsequent IKE negotiations and mandates how the peers are authenticated. You must create an IKE policy at each peer participating in the IKE negotiation.

If you do not configure any IKE policies, your router will use the default policy, which is always set to the lowest priority and contains the default value of each parameter.

After the two peers agree upon a policy, the security parameters of the policy are identified by an SA established at each peer, and these SAs apply to all subsequent IKE traffic during the negotiation.

You can configure multiple, prioritized policies on each peer—each with a different combination of parameter values. However, at least one of these policies must contain exactly the same encryption, hash, authentication, and Diffie-Hellman parameter values as one of the policies on the remote peer. For each policy that you create, you assign a unique priority (1 through 10,000, with 1 being the highest priority).

Configuring Advanced Encryption Standard in an IKE Policy Map

The Advanced Encryption Standard (AES) is a privacy transform for IPSec and Internet Key Exchange (IKE) that has been developed to replace the Data Encryption Standard (DES). AES is designed to be more secure than DES. AES offers a larger key size, while ensuring that the only known approach to decrypt a message is for an intruder to try every possible key. AES has a variable key length—the algorithm can specify a 128-bit key (the default), a 192-bit key, or a 256-bit key.

To configure the AES encryption algorithm within an IKE policy map, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp policy** *priority*	Defines an ISAKMP policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest.
Step 2	Router(config-isakmp)# **encryption** {**aes** | **aes 192** | **aes 256**} **...** Router(config-isakmp) # **exit**	Specifies the encryption algorithm within an IKE policy. • **aes**—Specifies 128-bit AES as the encryption algorithm. • **aes 192**—Specifies 192-bit AES as the encryption algorithm. • **aes 256**—Specifies 256-bit AES as the encryption algorithm. Specify any other policy values appropriate to your configuration, and then exit ISAKMP policy configuration mode. For details on configuring an ISAKMP policy, see the *Cisco IOS Security Configuration Guide*.

Verifying the AES IKE Policy

To verify the configuration of the AES IKE policy, enter the **show crypto isakmp policy** command:

```
Router# show crypto isakmp policy

Protection suite of priority 1
encryption algorithm:  AES - Advanced Encryption Standard (256 bit keys).
hash algorithm:        Secure Hash Standard
authentication method: Pre-Shared Key
Diffie-Hellman group:  #1 (768 bit)
lifetime: 3600 seconds, no volume limit
```

For an AES configuration example, see the "Advanced Encryption Standard Configuration Example" section on page 24-23.

Configuring ISAKMP Keyrings

The ISAKMP Keyrings feature (also known as the SafeNet IPSec VPN Client Support feature) allows you to limit the scope of an Internet Security Association and Key Management Protocol (ISAKMP) profile or ISAKMP keyring configuration to a local termination address or interface. The benefit of this feature is that different customers can use the same peer identities and ISAKMP keys by using different local termination addresses.

ISAKMP Keyrings Configuration Guidelines and Restrictions

When configuring ISAKMP keyrings, follow these guidelines and restrictions:

- The local address option works only for the primary address of an interface.
- If an IP address is provided, the administrator must ensure that the connection of the peer terminates to the address that is provided.
- If the IP address does not exist on the device, or if the interface does not have an IP address, the ISAKMP profile or ISAKMP keyring will be effectively disabled.

Limiting an ISAKMP Profile to a Local Termination Address or Interface

To configure an ISAKMP profile and limit it to a local termination address or interface, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp profile** *profile-name*	Defines an ISAKMP profile and enters ISAKMP profile configuration mode. • *profile-name*—Name of the ISAKMP profile.
Step 2	Router(conf-isa-profile)# **keyring** *keyring-name*	(Optional) Configures a keyring with an ISAKMP profile. • *keyring-name*—Name of the crypto keyring. **Note** A keyring is not needed inside an ISAKMP profile for local termination to work. Local termination works even if Rivest, Shamir, and Adelman (RSA) certificates are used.
Step 3	Router(conf-isa-profile)# **match identity address** *address*	Matches an identity from a peer in an ISAKMP profile. • *address*—IP address of the remote peer.
Step 4	Router(conf-isa-profile)# **local-address** {*interface-name* | *ip-address* [*vrf-tag*]}	Limits the scope of an ISAKMP profile or an ISAKMP keyring configuration to a local termination address or interface. • *interface-name*—Name of the local interface. • *ip-address*—Local termination address. • *vrf-tag*—(Optional) Scope of the IP address will be limited to the VRF.

Limiting a Keyring to a Local Termination Address or Interface

To configure an ISAKMP keyring and limit its scope to a local termination address or interface, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **keyring** *keyring-name*	Defines a crypto keyring to be used during IKE authentication and enters keyring configuration mode. • *keyring-name*—Name of the crypto keyring.
Step 2	Router(conf-keyring)# **local-address** {*interface-name* | *ip-address* [*vrf-tag*]}	Limits the scope of an ISAKMP profile or an ISAKMP keyring configuration to a local termination address or interface. • *interface-name*—Name of the local interface. • *ip-address*—Local termination address. • *vrf-tag*—(Optional) Scope of the IP address will be limited to the VRF.
Step 3	Router(conf-keyring)# **pre-shared-key address** *address*	Defines a preshared key to be used for IKE authentication. • *address*—IP address.

For complete configuration information for SafeNet IPSec VPN Client support, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_14/gt_scse.htm

For ISAKMP keyrings configuration examples, see the "ISAKMP Keyrings Configuration Examples" section on page 24-23.

Configuring Certificate to ISAKMP Profile Mapping

The Certificate to ISAKMP Profile Mapping feature enables you to assign an Internet Security Association and Key Management Protocol (ISAKMP) profile to a peer on the basis of the contents of arbitrary fields in the certificate. In addition, this feature allows you to assign a group name to those peers that are assigned an ISAKMP profile.

Note Certificate to ISAKMP Profile Mapping is only supported as of Cisco IOS Release 12.2(33)SRA.

Certificate to ISAKMP Profile Mapping Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Certificate to ISAKMP Profile Mapping:

- This feature will not be applicable if you use Rivest, Shamir, and Adelman (RSA)- signature or RSA-encryption authentication without certificate exchange. ISAKMP peers must be configured to do RSA-signature or RSA-encryption authentication using certificates.

Mapping the Certificate to the ISAKMP Profile

To map the certificate to the ISAKMP profile, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp profile** *profile-name*	Defines an ISAKMP profile and enters ISAKMP profile configuration mode • *profile-name*—Name of the user profile.
Step 2	Router(config-isa-prof)# **match certificate** *certificate-map*	Accepts the name of a certificate map. • *certificate-map*—Name of the certificate map.

Verifying the Certificate to ISAKMP Profile Mapping Configuration

To verify that the subject name of the certificate map has been properly configured, enter the **show crypto pki certificates** and the **debug crypto isakmp** commands.

The **show crypto pki certificates** command displays all current IKE security associations (SAs) at a peer. The **debug crypto isakmp** command displays messages about IKE events.

The following examples show that a certificate has been mapped to an ISAKMP profile. The examples include the configurations for the responder and initiator, the **show crypto pki certificates** command output verifying that the subject name of the certificate map has been configured, and **the debug crypto isakmp** command output showing that the certificate has gone through certificate map matching and been matched to the ISAKMP profile.

Responder Configuration

```
crypto pki certificate map cert_map 10
! The above line is the certificate map definition.
 subject-name co ou = green
! The above line shows that the subject name must have "ou = green."
!
crypto isakmp profile certpro
! The above line shows that this is the ISAKMP profile that will match if the certificate
of the peer matches cert_map (shown on third line below).
   ca trust-point 2315
   ca trust-point LaBcA
   match certificate cert_map
   initiate mode aggressive
```

Initiator Configuration

```
crypto ca trustpoint LaBcA
 enrollment url http://10.76.82.20:80/cgi-bin/openscep
 subject-name ou=green,c=IN
! The above line ensures that the subject name "ou = green" is set.
 revocation-check none
```

Command Output for show crypto pki certificates for the Initiator

```
Router# show crypto pki certificates
Certificate
  Status: Available
  Certificate Serial Number: 21
```

```
  Certificate Usage: General Purpose
  Issuer:
    cn=blue-lab CA
    o=CISCO
    c=IN
  Subject:
    Name: Router1.cisco.com
    c=IN
    ou=green
! The above line is a double check that "ou = green" has been set as the subject name.
    hostname=Router1.cisco.com
  Validity Date:
    start date: 14:34:30 UTC Mar 31 2004
    end   date: 14:34:30 UTC Apr 1 2009
    renew date: 00:00:00 UTC Jan 1 1970
  Associated Trustpoints: LaBcA
```

Command Output for debug crypto isakmp for the Responder

```
Router# debug crypto isakmp
6d23h: ISAKMP (0:268435460): received packet from 192.0.0.2 dport 500 sport 500 Global (R)
MM_KEY_EXCH
6d23h: ISAKMP: Main Mode packet contents (flags 1, len 892):
6d23h:           ID payload
6d23h:             FQDN <Router1.cisco.com> port 500 protocol 17
6d23h:           CERT payload
6d23h:           SIG payload
6d23h:           KEEPALIVE payload
6d23h:           NOTIFY payload
6d23h: ISAKMP:(0:4:HW:2):Input = IKE_MESG_FROM_PEER, IKE_MM_EXCH
6d23h: ISAKMP:(0:4:HW:2):Old State = IKE_R_MM4  New State = IKE_R_MM5
6d23h: ISAKMP:(0:4:HW:2): processing ID payload. message ID = 0
6d23h: ISAKMP (0:268435460): ID payload
        next-payload : 6
        type         : 2
        FQDN name    : Router1.cisco.com
        protocol     : 17
        port         : 500
        length       : 28
6d23h: ISAKMP:(0:4:HW:2):: peer matches *none* of the profiles
6d23h: ISAKMP:(0:4:HW:2): processing CERT payload. message ID = 0
6d23h: ISAKMP:(0:4:HW:2): processing a CT_X509_SIGNATURE cert
6d23h: ISAKMP:(0:4:HW:2): peer's pubkey isn't cached
6d23h: ISAKMP:(0:4:HW:2): OU = green
6d23h: ISAKMP:(0:4:HW:2): certificate map matches certpro profile
! The above line shows that the certificate has gone through certificate map matching and
that it matches the "certpro" profile.
6d23h: ISAKMP:(0:4:HW:2): Trying to re-validate CERT using new profile
6d23h: ISAKMP:(0:4:HW:2): Creating CERT validation list: 2315, LaBcA,
6d23h: ISAKMP:(0:4:HW:2): CERT validity confirmed.
```

Assigning the Group Name to the Peer

To associate a group name with an ISAKMP profile that will be assigned to a peer, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp profile** *profile-name*	Defines an ISAKMP profile and enters ISAKMP profile configuration mode • *profile-name*—Name of the user profile.
Step 2	Router (conf-isa-prof)# **client configuration group** *group-name*	Accepts the name of a group that will be assigned to a peer when the peer is assigned this crypto ISAKMP profile. • *group-name*—Name of the group to be associated with the peer.

Verifying the Group Name to Peer Assignation Configuration

To verify that a group has been assigned to a peer, enter the **debug crypto isakmp** command.

The **debug crypto isakmp** command displays messages about IKE events.

The following **debug crypto isakmp** output shows that the peer has been matched to the ISAKMP profile named “certpro” and that it has been assigned a group named “new_group.”

Initiator Configuration

```
crypto isakmp profile certpro
   ca trust-point 2315
   ca trust-point LaBcA
   match certificate cert_map
   client configuration group new_group
! The statement on the above line will assign the group "new_group" to any peer that
matches the ISAKMP profile "certpro."
   initiate mode aggressive
```

Command Output for debug crypto isakmp for the Responder

```
Router# debug crypto isakmp
6d23h: ISAKMP (0:268435461): received packet from 192.0.0.2 dport 500 sport 500 Global (R)
MM_KEY_EXCH
6d23h: ISAKMP: Main Mode packet contents (flags 1, len 892):
6d23h:           ID payload
6d23h:             FQDN <Router1.cisco.com> port 500 protocol 17
6d23h:           CERT payload
6d23h:           SIG payload
6d23h:           KEEPALIVE payload
6d23h:           NOTIFY payload
6d23h: ISAKMP:(0:5:HW:2):Input = IKE_MESG_FROM_PEER, IKE_MM_EXCH
6d23h: ISAKMP:(0:5:HW:2):Old State = IKE_R_MM4  New State = IKE_R_MM5
6d23h: ISAKMP:(0:5:HW:2): processing ID payload. message ID = 0
6d23h: ISAKMP (0:268435461): ID payload
        next-payload : 6
        type         : 2
        FQDN name    : Router1.cisco.com
        protocol     : 17
        port         : 500
```

```
        length       : 28
6d23h: ISAKMP:(0:5:HW:2):: peer matches *none* of the profiles
6d23h: ISAKMP:(0:5:HW:2): processing CERT payload. message ID = 0
6d23h: ISAKMP:(0:5:HW:2): processing a CT_X509_SIGNATURE cert
6d23h: ISAKMP:(0:5:HW:2): peer's pubkey isn't cached
6d23h: ISAKMP:(0:5:HW:2): OU = green
6d23h: ISAKMP:(0:5:HW:2): certificate map matches certpro profile
6d23h: ISAKMP:(0:5:HW:2): Trying to re-validate CERT using new profile
6d23h: ISAKMP:(0:5:HW:2): Creating CERT validation list: 2315, LaBcA,
6d23h: ISAKMP:(0:5:HW:2): CERT validity confirmed.
6d23h: ISAKMP:(0:5:HW:2):Profile has no keyring, aborting key search
6d23h: ISAKMP:(0:5:HW:2): Profile certpro assigned peer the group named new_group
```

For complete configuration information for certificate to ISAKMP profile mapping, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_8/gt_isakp.htm

For certificate to ISAKMP profile mapping configuration examples, see the "Certificate to ISAKMP Profile Mapping Configuration Examples" section on page 24-24.

Configuring an Encrypted Preshared Key

The Encrypted Preshared Key feature allows you to securely store plain text passwords in type 6 (encrypted) format in NVRAM.

Encrypted Preshared Key Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring an encrypted preshared key:

- Old ROM monitors (ROMmons) and boot images cannot recognize the new type 6 passwords. Therefore, errors are expected if you boot from an old ROMmon.
- If the password (master key) is changed, or re-encrypted, using the **key config-key password-encryption** command, the list registry passes the old key and the new key to the application modules that are using type 6 encryption.
- If the master key that was configured using the **key config-key password-encryption** command is deleted from the system, a warning is printed (and a confirm prompt is issued) that states that all type 6 passwords will become useless. As a security measure, after the passwords have been encrypted, they will never be decrypted in the Cisco IOS software. However, passwords can be re-encrypted.

Caution If the password configured using the **key config-key password-encryption** command is lost, it cannot be recovered. The password should be stored in a safe location.

- If you later unconfigure password encryption using the **no password encryption ae**s command, all existing type 6 passwords are left unchanged, and as long as the password (master key) that was configured using the **key config-key password-encryption** command exists, the type 6 passwords will be decrypted as and when required by the application.
- Because no one can "read" the password (configured using the **key config-key password-encryption** command), there is no way that the password can be retrieved from the router. Existing management stations cannot "know" what it is unless the stations are enhanced to include this key somewhere, in which case the password needs to be stored securely within the management system. If configurations are stored using TFTP, the configurations are not standalone,

meaning that they cannot be loaded onto a router. Before or after the configurations are loaded onto a router, the password must be manually added (using the **key config-key password-encryption** command). The password can be manually added to the stored configuration but is not recommended because adding the password manually allows anyone to decrypt all passwords in that configuration.

- If you enter or cut and paste cipher text that does not match the master key, or if there is no master key, the cipher text is accepted or saved, but the following alert message is printed:

```
ciphertext>[for username bar>] is incompatible with the configured master key
```

- If a new master key is configured, all the plain keys are encrypted and made type 6 keys. The existing type 6 keys are not encrypted. The existing type 6 keys are left as is.
- If the old master key is lost or unknown, you have the option of deleting the master key using the **no key config-key password-encryption** command. Deleting the master key using the **no key config-key password-encryption** command causes the existing encrypted passwords to remain encrypted in the router configuration. The passwords will not be decrypted.

Configuring an Encrypted Preshared Key

To configure an encrypted preshared key, perform the following steps beginning global configuration mode:

	Command	Purpose

Step 1	Router(config)# **key config-key password-encryption**	Stores a type 6 encryption key in private NVRAM.
		Note the following:
		• If you are entering the key interactively (using the **Enter** key) and an encrypted key already exists, you will be prompted for the following:
		`Old key, New key, and Confirm key`
		• If you are entering the key interactively but an encryption key is not present, you will be prompted for the following:
		`New key and Confirm key`
		• If you are removing a password that is already encrypted, you will see the following prompt:
		`WARNING: All type 6 encrypted keys will become unusable. Continue with master key deletion? [yes/no]:`

Step 2	Router(config)# **password-encryption aes**	Enables the encrypted preshared key.

Verifying the Encrypted Preshared Key Configuration

To verify that a new master key has been configured and that the keys have been encrypted with the new master key, enter the **password logging** command. The following is an example of its output:

```
Router# password logging

Router (config)# key config-key password-encrypt

New key:
Confirm key:
Router (config)#
01:40:57: TYPE6_PASS: New Master key configured, encrypting the keys with
the new master keypas

Router (config)# key config-key password-encrypt
Old key:
New key:
Confirm key:
Router (config)#
01:42:11: TYPE6_PASS: Master key change heralded, re-encrypting the keys
with the new master key
01:42:11: TYPE6_PASS: Mac verification successful
01:42:11: TYPE6_PASS: Mac verification successful
01:42:11: TYPE6_PASS: Mac verification successful
```

For complete configuration information for the Encrypted Preshared Key feature, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_2/gt_epsk.htm

For an encrypted preshared key configuration example, see the "Encrypted Preshared Key Configuration Example" section on page 24-25.

Configuring IKE Aggressive Mode Initiation

The IKE Aggressive Mode Initiation feature allows you to configure Internet Key Exchange (IKE) preshared keys as RADIUS tunnel attributes for IP Security (IPSec) peers.

Although IKE preshared keys are easy to deploy, they do not scale well with an increasing number of users and are therefore prone to security threats. Instead of keeping your preshared keys on the hub router, this feature allows you to scale your preshared keys by storing and retrieving them from an authentication, authorization, and accounting (AAA) server. The preshared keys are stored in the AAA server as Internet Engineering Task Force (IETF) RADIUS tunnel attributes and are retrieved when a user tries to "speak" to the hub router. The hub router retrieves the preshared key from the AAA server and the spokes (the users) initiate aggressive mode to the hub by using the preshared key that is specified in the Internet Security Association and Key Management Protocol (ISAKMP) peer policy as a RADIUS tunnel attribute.

IKE Aggressive Mode Initiation Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring IKE Aggressive Mode Initiation:

- IKE Aggressive Mode Initiation is not intended to be used with a dynamic crypto map that uses Tunnel Endpoint Discovery (TED) to initiate tunnel setup. TED is useful in configuring a full-mesh setup, which requires an AAA server at each site to store the preshared keys for the peers; this configuration is not practical for use with this feature.
- Only the following ID types can be used in this feature:
  - ID_IPV4 (IPV4 address)
  - ID_FQDN (fully qualified domain name, for example, "foo.cisco.com")
  - ID_USER_FQDN (e-mail address)
- Before configuring IKE Aggressive Mode Initiation, you must perform the following tasks:
  - Configure AAA
  - Configure an IPSec transform
  - Configure a static crypto map
  - Configure an ISAKMP policy
  - Configure a dynamic crypto map

  For information on completing these tasks, refer to the *Cisco IOS Security Configuration Guide*.

To configure IKE Aggressive Mode Initiation, you must configure the Tunnel-Client-Endpoint and Tunnel-Password attributes within the ISAKMP peer configuration. To do this, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto map** *map-name* **isakmp authorization list** *list-name*	Enables IKE querying of AAA for tunnel attributes in aggressive mode. • *map-name*—Name you assign to the crypto map set. • *list-name*—Character string used to name the list of authorization methods activated when a user logs in. The list name must match the list name defined during AAA configuration.
Step 2	Router(config)# **crypto isakmp peer** {**ip-address** *ip-address* | **fqdn** *fqdn*}	Enables an IPSec peer for IKE querying of AAA for tunnel attributes in aggressive mode and enters ISAKMP policy configuration mode. • *ip-address*—IP address of the peer router. • *fqdn*—Fully-qualified domain name of the peer router.

	Command	Purpose
Step 3	Router(config-isakmp)# **set aggressive-mode client-endpoint** *client-endpoint*	Specifies the Tunnel-Client-Endpoint attribute within an ISAKMP peer configuration. • *client-endpoint*—One of the following ID types of the initiator end of the tunnel: – ID_IPV4 (IPV4 address) – ID_FQDN (fully qualified domain name, for example, “foo.cisco.com”) – ID_USER_FQDN (e-mail address) **Note** The ID type is translated to the corresponding ID type in Internet Key Exchange (IKE).
Step 4	Router(config-isakmp)# **set aggressive-mode password** *password*	Specifies the Tunnel-Password attribute within an ISAKMP peer configuration. • *password*—Password that is used to authenticate the peer to a remote server. The tunnel password is used as the Internet Key Exchange (IKE) preshared key.

Verifying the IKE Aggressive Mode Initiation Configuration

To verify that the Tunnel-Client-Endpoint and Tunnel-Password attributes have been configured within the ISAKMP peer policy, use the **show running-config** global configuration command.

For complete configuration information for IKE Aggressive Mode Initiation, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/ft_ikeag.htm

For IKE Aggressive Mode Initiation configuration examples, see the “IKE Aggressive Mode Initiation Configuration Examples” section on page 24-25.

Configuring Call Admission Control for IKE

Call Admission Control for IKE allows you to limit the number of simultaneous IKE security associations (SAs) that a router can establish.

Note Call Admission Control is only supported as of Cisco IOS Release 12.2(33)SRA.

There are two ways to limit the number of IKE SAs that a router can establish to or from another router:

- Configure an absolute IKE SA limit by entering the **crypto call admission limit** command. When an IKE SA limit is defined, the router drops new IKE SA requests when this value has been reached as follows: When there is a new SA request from a peer router, IKE determines if the number of active IKE SAs plus the number of SAs being negotiated meets or exceeds the configured SA limit. If the number is greater than or equal to the limit, the new SA request is rejected and a syslog is generated. This log contains the source destination IP address of the SA request.

- Configure a system resource limit by entering the **call admission limit** command. When a system resource limit is defined, the router drops new IKE SA requests when the specified percentage of system resources is being used as follows: Call Admission Control (CAC) polls a global resource monitor so that IKE knows when the router is running short of CPU cycles or memory buffers. You can configure a resource limit, from 1 to 100, that represents a percentage of system resources. When that percentage of the system resources is being used, IKE drops (will not accept new) SA requests. For example, if you specify a resource limit of 90 percent, IKE stops accepting SA requests when 90 percent of the system resources is being used.

CAC is applied only to new SAs (that is, when an SA does not already exist between the peers). Every effort is made to preserve existing SAs. Only new SA requests will ever be denied due to a lack of system resources or because the configured IKE SA limit has been reached.

Configuring the IKE Security Association Limit

To configure an IKE Security Association limit, perform the following steps beginning in global configuration mode. When an IKE SA limit is defined, the router drops new IKE SA requests when the limit has been reached:

	Command	Purpose
Step 1	Router(config)# **crypto call admission limit ike sa** *number*	Specifies the maximum number of IKE SAs that the router can establish before IKE begins rejecting new SA requests. • *number*—Number of active IKE SAs allowed on the router. The value must be greater than 1.
Step 2	Router (config)# **exit**	Returns to privileged EXEC mode.

Configuring a System Resource Limit

To configure a system resource limit, perform the following steps beginning in global configuration mode. When an IKE SA limit is defined, the router drops new IKE SA requests when the specified percentage of system resources is being used.

	Command	Purpose
Step 1	Router(config)# **call admission limit** *percent*	Instructs IKE to stop accepting new SA requests (that is, calls for CAC) when the specified percentage of system resources is being used. • *percent*—Percentage of the system resources that, when used, causes IKE to stop accepting new SA requests. Valid values are 1 to 100.
Step 2	Router (config)# **exit**	Returns to privileged EXEC mode.

Clearing Call Admission Statistics

To clear the Call Admission Control counters that track the number of accepted and rejected Internet Key Exchange (IKE) requests, use the **clear crypto call admission statistics** command in global configuration mode:

Router(config)# **clear crypto call admission statistics**

Verifying the Call Admission Control for IKE Configuration

To verify that Call Admission Control has been configured, enter the **show call admission statistics** and the **show crypto call admission statistics** commands.

The **show call admission statistics** command monitors the global CAC configuration parameters and the behavior of CAC.

```
Router# show call admission statistics
Total Call admission charges: 0, limit 25
Total calls rejected 12, accepted 51
Load metric: charge 0, unscaled 0
```

The **show crypto call admission statistics** command monitors crypto CAC statistics.

```
Router# show crypto call admission statistics
-----------------------------------------------------------
               Crypto Call Admission Control Statistics
-----------------------------------------------------------
System Resource Limit: 0    Max IKE SAs 0
Total IKE SA Count:    0    active:     0   negotiating: 0
Incoming IKE Requests: 0    accepted:   0   rejected:    0
Outgoing IKE Requests: 0    accepted:   0   rejected:    0
Rejected IKE Requests: 0    rsrc low:   0   SA limit:    0
```

For more complete configuration information for Call Admission Control for IKE, refer to the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_8/gtcallik.htm

For Call Admission Control for IKE configuration examples, see the "Call Admission Control for IKE Configuration Examples" section on page 24-26.

Configuring Dead Peer Detection

Dead Peer Detection (DPD), defined in RFC 3706, is a mechanism used to detect dead IPSec peers. IPSec is a peer-to-peer type of technology. It is possible that IP connectivity may be lost between peers due to routing problems, peer reloading, or some other situation. This lost connectivity can result in black holes where traffic is lost. DPD, based on a traffic-detection method, is one possible mechanism to remedy this situation.

Note The **periodic** option of the **crypto isakmp keepalive** command is only supported as of Cisco IOS Release 12.2(33)SRA; the **on-demand** option is supported in all releases.

DPD supports two options: on-demand or periodic. The on-demand approach is the default. With on-demand DPD, messages are sent on the basis of traffic patterns. For example, if a router must send outbound traffic and the liveliness of the peer is questionable, the router sends a DPD message to query the status of the peer. If a router has no traffic to send, it never sends a DPD message. If a peer is dead, and the router never has any traffic to send to the peer, the router will not find out until the IKE or IPSec security association (SA) has to be rekeyed (the liveliness of the peer is unimportant if the router is not trying to communicate with the peer). On the other hand, if the router has traffic to send to the peer, and the peer does not respond, the router will initiate a DPD message to determine the state of the peer.

With the periodic option, you can configure your router so that DPD messages are "forced" at regular intervals. This forced approach results in earlier detection of dead peers. For example, if a router has no traffic to send, a DPD message is still sent at regular intervals, and if a peer is dead, the router does not have to wait until the IKE SA times out to find out.

DPD is configured using the **crypto isakmp keepalive** command. DPD and Cisco IOS keepalives function on the basis of a timer. If the timer is set for 10 seconds, the router will send a "hello" message every 10 seconds (unless, of course, the router receives a "hello" message from the peer). The benefit of Cisco IOS keepalives and periodic DPD is earlier detection of dead peers. However, Cisco IOS keepalives and periodic DPD rely on periodic messages that have to be sent with considerable frequency. The result of sending frequent messages is that the communicating peers must encrypt and decrypt more packets.

DPD and Cisco IOS keepalive features can be used in conjunction with multiple peers in the crypto map to allow for stateless failover. DPD allows the router to detect a dead IKE peer, and when the router detects the dead state, the router deletes the IPSec and IKE SAs to the peer. If you configure multiple peers, the router will switch over to the next listed peer for a stateless failover.

Dead Peer Detection Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring DPD:

- If you do not configure the **periodic** option using the **crypto isakmp keepalive** command, the router defaults to the **on-demand** approach.
- Before configuring periodic DPD, you should ensure that your IKE peer supports DPD. Implementations that support DPD include the Cisco VPN 3000 concentrator, Cisco PIX Firewall, Cisco VPN Client, and Cisco IOS software in all modes of operation—site-to-site, Easy VPN remote, and Easy VPN server.
- Using periodic DPD potentially allows the router to detect an unresponsive IKE peer with better response time when compared to on-demand DPD. However, use of periodic DPD incurs extra overhead. When communicating to large numbers of IKE peers, you should consider using on-demand DPD instead.
- When the **crypto isakmp keepalive** command is configured, the Cisco IOS software negotiates the use of Cisco IOS keepalives or DPD, depending on which protocol the peer supports.

Configuring a Dead Peer Detection Message

To allow the router to send DPD messages to the peer, enter the **crypto isakmp keepalive** command in global configuration mode as follows:

Router(config)# **crypto isakmp keepalive** *seconds* [*retries*] [**periodic | on-demand**]

In this command:

- *seconds* specifies the number of seconds between DPD messages; the range is from 10 to 3600 seconds.
- *retries* (Optional) specifies the number of seconds between DPD retries if the DPD message fails; the range is from 2 to 60 seconds. If unspecified, the default is 2 seconds.
- **periodic** (Optional) specifies the that DPD messages are sent at regular intervals.
- **on-demand** (Optional) specifies DPD retries are sent on demand. This is the default behavior.

Note Because the **on-demand** option is the default, the **on-demand** keyword does not appear in configuration output.

Configuring Dead Peer Detection and Cisco IOS Keepalives with Multiple Peers in a Crypto Map

To configure DPD and Cisco IOS keepalives to be used in conjunction with the crypto map to allow for stateless failover, perform the following steps beginning in global configuration mode. This configuration will cause a router to cycle through the peer list when it detects that the first peer is dead:

	Command	Purpose
Step 1	Router(config)# **crypto map** *map-name seq-num* **ipsec-isakmp**	Enters crypto map configuration modes and creates or modifies a crypto map entry. • *map-name*—Name that identifies the map set. • *seq-num*—Sequence number assigned to the crypto map entry. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec SAs for protecting the traffic specified by this crypto map entry.
Step 2	Router(config-crypto-map)# **set peer** {*host-name* [**dynamic**] |*ip-address*}	Specifies an IPSec peer in a crypto map entry. • *hostname*—IPSec peer host name. • **dynamic**—(Optional) Indicates that the host name of the IPSec peer will be resolved via a domain name server (DNS) lookup right before the router establishes the IPSec tunnel. • *ip-address*—IPSec peer IP address. You can specify multiple peers by repeating this command.

	Command	Purpose
Step 3	Router(config-crypto-map)# **set transform-set** *transform-set-name*	Specifies which transform sets can be used with the crypto map entry. • *transform-set-name*—Name of the transform set. You can specify more than one transform set name by repeating this command.
Step 4	Router(config-crypto-map)# **match address** [*access-list-id* | *name*]	Specifies an extended access list for a crypto map entry. • *access-list-id*—(Optional) Identifies the extended access list by its name or number. This value should match the access-list-number or name argument of the extended access list being matched. • *name*—(Optional) Identifies the named encryption access list. This name should match the name argument of the named encryption access list being matched.

Verifying the Dead Peer Detection Configuration

To verify that DPD is enabled (and that the peer supports DPD), enter the **debug crypto isakmp** command:

```
Router(config)# debug crypto isakmp

*Mar 25 15:17:14.131: ISAKMP:(0:1:HW:2):IKE_DPD is enabled, initializing timers
```

You should see debug messages at the interval specified by the command.

The following example corresponds to sending the DPD R_U_THERE message:

```
*Mar 25 15:18:52.107: ISAKMP:(0:1:HW:2): sending packet to 10.2.80.209 my_port
500 peer_port 500 (I) QM_IDLE
*Mar 25 15:18:52.107: ISAKMP:(0:1:HW:2):purging node 899852982 *Mar 25 15:18:52.111:
ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_TIMER,
IKE_TIMER_IM_ALIVE
*Mar 25 15:18:52.111: ISAKMP:(0:1:HW:2):Old State = IKE_P1_COMPLETE New State =
IKE_P1_COMPLETE
```

The following example corresponds to receiving the acknowledge (ACK) message from the peer:

```
*Mar 25 15:18:52.123: ISAKMP (0:268435457): received packet from 10.2.80.209
dport 500 sport 500 Global (I) QM_IDLE
*Mar 25 15:18:52.123: ISAKMP: set new node -443923643 to QM_IDLE *Mar 25 15:18:52.131:
ISAKMP:(0:1:HW:2): processing HASH payload. message ID =
-443923643
*Mar 25 15:18:52.131: ISAKMP:(0:1:HW:2): processing NOTIFY R_U_THERE_ACK protocol 1
spi 0, message ID = -443923643, sa = 81BA4DD4
*Mar 25 15:18:52.135: ISAKMP:(0:1:HW:2): DPD/R_U_THERE_ACK received from peer
10.2.80.209, sequence 0x9
*Mar 25 15:18:52.135: ISAKMP:(0:1:HW:2):deleting node -443923643 error FALSE
reason "informational (in) state 1"
*Mar 25 15:18:52.135: ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_PEER, IKE_INFO_NOTIFY *Mar
25 15:18:52.135: ISAKMP:(0:1:HW:2):Old State = IKE_P1_COMPLETE New State =
IKE_P1_COMPLETE
```

The following example shows what happens when the remote peer is unreachable. The router sends one DPD R_U_THERE message and four retransmissions before it finally deletes the IPSec and IKE SAs:

```
Router#
*Mar 25 15:47:35.335: ISAKMP: set new node -90798077 to QM_IDLE *Mar 25 15:47:35.343:
ISAKMP:(0:1:HW:2): sending packet to 10.2.80.209 my_port
500 peer_port 500 (I) QM_IDLE
*Mar 25 15:47:35.343: ISAKMP:(0:1:HW:2):purging node -90798077 *Mar 25 15:47:35.347:
ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_TIMER,
IKE_TIMER_IM_ALIVE
*Mar 25 15:47:35.347: ISAKMP:(0:1:HW:2):Old State = IKE_P1_COMPLETE New State =
IKE_P1_COMPLETE
*Mar 25 15:47:36.611: ISAKMP:(0:1:HW:2):purging node 1515050537 *Mar 25 15:47:37.343:
ISAKMP:(0:1:HW:2):incrementing error counter on sa:
PEERS_ALIVE_TIMER
*Mar 25 15:47:37.343: ISAKMP: set new node -1592471565 to QM_IDLE *Mar 25 15:47:37.351:
ISAKMP:(0:1:HW:2): sending packet to 10.2.80.209 my_port
500 peer_port 500 (I) QM_IDLE
*Mar 25 15:47:37.351: ISAKMP:(0:1:HW:2):purging node -1592471565 *Mar 25 15:47:37.355:
ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_TIMER,
IKE_TIMER_PEERS_ALIVE
*Mar 25 15:47:37.355: ISAKMP:(0:1:HW:2):Old State = IKE_P1_COMPLETE New State =
IKE_P1_COMPLETE
*Mar 25 15:47:39.355: ISAKMP:(0:1:HW:2):incrementing error counter on sa:
PEERS_ALIVE_TIMER
*Mar 25 15:47:39.355: ISAKMP: set new node 1758739401 to QM_IDLE *Mar 25 15:47:39.363:
ISAKMP:(0:1:HW:2): sending packet to 10.2.80.209 my_port
500 peer_port 500 (I) QM_IDLE
*Mar 25 15:47:39.363: ISAKMP:(0:1:HW:2):purging node 1758739401 *Mar 25 15:47:39.367:
ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_TIMER,
IKE_TIMER_PEERS_ALIVE
*Mar 25 15:47:39.367: ISAKMP:(0:1:HW:2):Old State = IKE_P1_COMPLETE New State =
IKE_P1_COMPLETE
*Mar 25 15:47:41.367: ISAKMP:(0:1:HW:2):incrementing error counter on sa:
PEERS_ALIVE_TIMER
*Mar 25 15:47:41.367: ISAKMP: set new node 320258858 to QM_IDLE *Mar 25 15:47:41.375:
ISAKMP:(0:1:HW:2): sending packet to 10.2.80.209 my_port
500 peer_port 500 (I) QM_IDLE
*Mar 25 15:47:41.379: ISAKMP:(0:1:HW:2):purging node 320258858 *Mar 25 15:47:41.379:
ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_TIMER,
IKE_TIMER_PEERS_ALIVE
*Mar 25 15:47:41.379: ISAKMP:(0:1:HW:2):Old State = IKE_P1_COMPLETE New State =
IKE_P1_COMPLETE
*Mar 25 15:47:43.379: ISAKMP:(0:1:HW:2):incrementing error counter on sa:
PEERS_ALIVE_TIMER
*Mar 25 15:47:43.379: ISAKMP: set new node -744493014 to QM_IDLE *Mar 25 15:47:43.387:
ISAKMP:(0:1:HW:2): sending packet to 10.2.80.209 my_port
500 peer_port 500 (I) QM_IDLE
*Mar 25 15:47:43.387: ISAKMP:(0:1:HW:2):purging node -744493014 *Mar 25 15:47:43.391:
ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_TIMER,
IKE_TIMER_PEERS_ALIVE
*Mar 25 15:47:43.391: ISAKMP:(0:1:HW:2):Old State = IKE_P1_COMPLETE New State =
IKE_P1_COMPLETE
*Mar 25 15:47:45.391: ISAKMP:(0:1:HW:2):incrementing error counter on sa:
PEERS_ALIVE_TIMER
*Mar 25 15:47:45.391: ISAKMP:(0:1:HW:2):peer 10.2.80.209 not responding! *Mar 25
15:47:45.391: ISAKMP:(0:1:HW:2):peer does not do paranoid keepalives.
*Mar 25 15:47:45.391: ISAKMP:(0:1:HW:2):deleting SA reason "peers alive" state
(I) QM_IDLE (peer 10.2.80.209) input queue 0
*Mar 25 15:47:45.395: ISAKMP: Unlocking IPSEC struct 0x81E5C4E8 from
delete_siblings, count 0
*Mar 25 15:47:45.395: %CRYPTO-5-SESSION_STATUS: Crypto tunnel is DOWN. Peer
10.2.80.209:500 Id: 10.2.80.209
```

```
*Mar 25 15:47:45.399: ISAKMP: set new node -2061951065 to QM_IDLE *Mar 25 15:47:45.411:
ISAKMP:(0:1:HW:2): sending packet to 10.2.80.209 my_port
500 peer_port 500 (I) QM_IDLE
*Mar 25 15:47:45.411: ISAKMP:(0:1:HW:2):purging node -2061951065 *Mar 25 15:47:45.411:
ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_TIMER,
IKE_TIMER_PEERS_ALIVE
*Mar 25 15:47:45.411: ISAKMP:(0:1:HW:2):Old State = IKE_P1_COMPLETE New State =
IKE_DEST_SA
*Mar 25 15:47:45.415: ISAKMP:(0:1:HW:2):deleting SA reason "peers alive" state
(I) QM_IDLE (peer 10.2.80.209) input queue 0
*Mar 25 15:47:45.415: ISAKMP: Unlocking IKE struct 0x81E5C4E8 for
isadb_mark_sa_deleted(), count 0
*Mar 25 15:47:45.415: ISAKMP: Deleting peer node by peer_reap for 10.2.80.209:
81E5C4E8
*Mar 25 15:47:45.415: ISAKMP:(0:1:HW:2):deleting node -1067612752 error TRUE
reason "peers alive"
*Mar 25 15:47:45.415: ISAKMP:(0:1:HW:2):deleting node -114443536 error TRUE
reason "peers alive"
*Mar 25 15:47:45.419: ISAKMP:(0:1:HW:2):deleting node 2116015069 error TRUE
reason "peers alive"
*Mar 25 15:47:45.419: ISAKMP:(0:1:HW:2):deleting node -1981865558 error TRUE
reason "peers alive"
*Mar 25 15:47:45.419: ISAKMP:(0:1:HW:2):Input = IKE_MESG_INTERNAL, IKE_PHASE1_DEL *Mar 25
15:47:45.419: ISAKMP:(0:1:HW:2):Old State = IKE_DEST_SA New State =
IKE_DEST_SA
*Mar 25 15:47:45.419: ISAKMP: received ke message (4/1)
*Mar 25 15:47:45.419: ISAKMP: received ke message (3/1)
*Mar 25 15:47:45.423: ISAKMP: ignoring request to send delete notify (no ISAKMP
sa) src 10.1.32.14 dst 10.2.80.209 for SPI 0x3A7B69BF
*Mar 25 15:47:45.423: ISAKMP:(0:1:HW:2):deleting SA reason "" state (I)
MM_NO_STATE (peer 10.2.80.209) input queue 0
*Mar 25 15:47:45.423: ISAKMP:(0:1:HW:2):deleting node -1067612752 error FALSE
reason ""
*Mar 25 15:47:45.423: ISAKMP:(0:1:HW:2):deleting node -114443536 error FALSE
reason ""
*Mar 25 15:47:45.423: ISAKMP:(0:1:HW:2):deleting node 2116015069 error FALSE
reason ""
*Mar 25 15:47:45.427: ISAKMP:(0:1:HW:2):deleting node -1981865558 error FALSE
reason ""
*Mar 25 15:47:45.427: ISAKMP:(0:1:HW:2):Input = IKE_MESG_FROM_PEER, IKE_MM_EXCH *Mar 25
15:47:45.427: ISAKMP:(0:1:HW:2):Old State = IKE_DEST_SA New State =
IKE_DEST_SA
```

For more complete configuration information for Cisco IOS Dead Peer Detection (DPD), refer to the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/gtdpmo.htm

For Dead Peer Detection configuration examples, see the "Dead Peer Detection Configuration Examples" section on page 24-27.

Configuring IPSec NAT Transparency

The IPSec NAT Transparency feature introduces support for IP Security (IPSec) traffic to travel through Network Address Translation (NAT) or Point Address Translation (PAT) points in the network by addressing many known incompatibilities between NAT and IPSec.

Before the introduction of this feature, a standard IPSec virtual private network (VPN) tunnel would not work if there were one or more NAT or PAT points in the delivery path of the IPSec packet. This feature makes NAT IPSec-aware, thereby allowing remote access users to build IPSec tunnels to home gateways.

IPSec NAT Transparency Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring IPSec NAT transparency:

- For non-DMVPN configurations, NAT transparency is supported in both tunnel and transport mode.
- For DMVPN configurations, NAT transparency is only supported in transport mode.

Configuring NAT Transparency

NAT transparency is a feature that is auto-detected by the IPSec VPN SPA. There are no configuration steps. If both VPN devices are NAT transparency-capable, NAT transparency is auto-detected and auto-negotiated.

Disabling NAT Transparency

You may wish to disable NAT transparency if you already know that your network uses IPSec-awareness NAT (SPI-matching scheme). To disable NAT transparency, use the following command in global configuration mode:

```
Router(config)# no crypto ipsec nat-transparency udp-encapsulation
```

Configuring NAT Keepalives

To configure your router to send NAT keepalive packets, enter the **crypto isakmp nat keepalive** command in global configuration mode:

```
Router(config)# crypto isakmp nat keepalive seconds
```

In this command, *seconds* specifies the number of seconds between keepalive packets; range is between 5 to 3,600 seconds.

Verifying the NAT Keepalives Configuration

To verify the NAT keepalives configuration, enter the **show crypto ipsec sa** command:

```
Router# show crypto ipsec sa

interface:GigabitEthernet5/0/1
    Crypto map tag:testtag, local addr. 10.2.80.161

    local ident (addr/mask/prot/port):(10.2.80.161/255.255.255.255/0/0)
    remote ident (addr/mask/prot/port):(100.0.0.1/255.255.255.255/0/0)
    current_peer:100.0.0.1:4500
    PERMIT, flags={origin_is_acl,}
    #pkts encaps:109, #pkts encrypt:109, #pkts digest 109
    #pkts decaps:109, #pkts decrypt:109, #pkts verify 109
    #pkts compressed:0, #pkts decompressed:0
    #pkts not compressed:0, #pkts compr. failed:0, #pkts decompress failed:0
    #send errors 90, #recv errors 0

    local crypto endpt.:10.2.80.161, remote crypto endpt.:100.0.0.1:4500
    path mtu 1500, media mtu 1500
```

```
current outbound spi:23945537

inbound esp sas:
spi:0xF423E273(4095992435)
transform:esp-des esp-sha-hmac ,
in use settings ={Tunnel UDP-Encaps, }
slot:0, conn id:200, flow_id:1, crypto map:testtag
sa timing:remaining key lifetime (k/sec):(4607996/2546)
IV size:8 bytes
replay detection support:Y

inbound ah sas:

inbound pcp sas:

outbound esp sas:
spi:0x23945537(596923703)
transform:esp-des esp-sha-hmac ,
in use settings ={Tunnel UDP-Encaps, }
slot:0, conn id:201, flow_id:2, crypto map:testtag
sa timing:remaining key lifetime (k/sec):(4607998/2519)
IV size:8 bytes
replay detection support:Y

outbound ah sas:

outbound pcp sas:
```

For complete configuration information for Cisco IOS IPSec NAT Transparency support, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ftipsnat.htm

For a NAT keepalives configuration example, see the "IPSec NAT Transparency Configuration Example" section on page 24-28.

Configuration Examples

This section provides examples of the following configurations:

Advanced Encryption Standard Configuration Example

The following example is sample output from the **show running-config** command. In this example, the Advanced Encryption Standard (AES) 256-bit key is enabled.

```
Router# show running-config

Current configuration : 1665 bytes
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname "Router1"
ip subnet-zero
no ip domain lookup
ip audit notify log
ip audit po max-events 100
crypto isakmp policy 10
encryption aes 256
authentication pre-share
lifetime 180
crypto isakmp key cisco123 address 10.0.110.1
crypto ipsec transform-set aesset esp-aes 256 esp-sha-hmac
mode transport
crypto map aesmap 10 ipsec-isakmp
set peer 10.0.110.1
set transform-set aesset
```

ISAKMP Keyrings Configuration Examples

The following examples show how to limit the scope of an Internet Security Association and Key Management Protocol (ISAKMP) profile or ISAKMP keyring configuration to a local termination address or interface:

- ISAKMP Profile Bound to a Local Interface Configuration Example, page 24-23
- ISAKMP Keyring Bound to a Local Interface Configuration Example, page 24-23
- ISAKMP Keyring Bound to a Local IP Address Configuration Example, page 24-24

ISAKMP Profile Bound to a Local Interface Configuration Example

The following example configures an ISAKMP profile bound to a local interface:

```
Router(config)# crypto isakmp profile profile1
Router(conf-isa-profile)# keyring keyring1
Router(conf-isa-profile)# match identity address 10.0.0.0 255.0.0.0
Router(conf-isa-profile)# local-address serial2/0
```

ISAKMP Keyring Bound to a Local Interface Configuration Example

The following example configures an ISAKMP keyring bound only to interface serial2/0:

```
Router(config)# crypto keyring keyring1
Router(conf-keyring)# local-address serial2/0
Router(conf-keyring)# pre-shared-key address 10.0.0.1
```

ISAKMP Keyring Bound to a Local IP Address Configuration Example

The following example configures an ISAKMP keyring bound only to IP address 10.0.0.2:

```
Router(config)# crypto keyring keyring1
Router(conf-keyring)# local-address 10.0.0.2
Router(conf-keyring)# pre-shared-key address 10.0.0.2 key
```

Certificate to ISAKMP Profile Mapping Configuration Examples

The following examples show how to configure Certificate to ISAKMP Profile Mapping:

- Certificates Mapped to the ISAKMP Profile on the Basis of Arbitrary Fields Configuration Example, page 24-24
- Group Name Assigned to a Peer That Is Associated with an ISAKMP Profile Configuration Example, page 24-24

Certificates Mapped to the ISAKMP Profile on the Basis of Arbitrary Fields Configuration Example

The following example shows that whenever a certificate contains "ou = green," the ISAKMP profile "cert_pro" will be assigned to the peer:

```
crypto pki certificate map cert_map 10
 subject-name co ou = green
!
crypto isakmp identity dn
crypto isakmp profile cert_pro
   ca trust-point 2315
   ca trust-point LaBcA
   initiate mode aggressive
   match certificate cert_map
```

Group Name Assigned to a Peer That Is Associated with an ISAKMP Profile Configuration Example

The following example shows that the group "some_group" is to be associated with a peer that has been assigned an ISAKMP profile:

```
crypto isakmp profile id_profile
   ca trust-point 2315
   match identity host domain cisco.com

client configuration group some_group
```

Encrypted Preshared Key Configuration Example

The following example shows a configuration for which a type 6 preshared key has been encrypted. It includes the prompts and messages that a user might see.

```
Router(config)# password encryption aes
Router(config)# key config-key password-encrypt
New key:
Confirm key:
Router(config)#
0:46:40: TYPE6_PASS: New Master key configured, encrypting the keys with
the new master key
Router(config)# exit
```

IKE Aggressive Mode Initiation Configuration Examples

This section provides the following IKE Aggressive Mode Initiation configuration examples:

- IKE Aggressive Mode Initiation Hub Configuration Example, page 24-25
- IKE Aggressive Mode Initiation Spoke Configuration Example, page 24-26
- IKE Aggressive Mode Initiation RADIUS User Profile Example, page 24-26

IKE Aggressive Mode Initiation Hub Configuration Example

The following example shows how to configure a hub for a hub-and-spoke topology that supports IKE aggressive mode initiation using RADIUS tunnel attributes:

```
!The AAA configurations are as follows:
aaa new-model
aaa authorization network ike group radius
aaa authentication login default group radius
!
! The Radius configurations are as follows:
radius-server host 1.1.1.1 auth-port 1645 acct-port 1646
radius-server key rad123
!
! The IKE configurations are as follows:
crypto isakmp policy 1
authentication pre-share
!
! The IPSec configurations are as follows:
crypto ipsec transform-set trans1 esp-3des esp-sha-hmac
!
crypto dynamic-map Dmap 10
set transform-set trans1
!
crypto map Testtag isakmp authorization list ike
crypto map Testtag 10 ipsec-isakmp dynamic Dmap
!
interface Ethernet0
ip address 4.4.4.1 255.255.255.0
crypto map Testtag
!
interface Ethernet1
ip address 2.2.2.1 255.255.255.0
```

IKE Aggressive Mode Initiation Spoke Configuration Example

The following example shows how to configure a spoke for a hub-and-spoke topology that supports IKE aggressive mode initiation using RADIUS tunnel attributes:

```
!The IKE configurations are as follows:
crypto isakmp policy 1
authentication pre-share
!
! The IPSec configurations are as follows:
crypto ipsec transform-set trans1 esp-3des esp-sha-hmac
access-list 101 permit ip 3.3.3.0 0.0.0.255 2.2.2.0 0.0.0.255
!
! Initiate aggressive mode using Radius tunnel attributes
crypto isakmp peer address 4.4.4.1
set aggressive-mode client-endpoint user-fqdn user@cisco.com
set aggressive-mode password cisco123
!
crypto map Testtag 10 ipsec-isakmp
set peer 4.4.4.1
set transform-set trans1
match address 101
!
interface Ethernet0
ip address 5.5.5.1 255.255.255.0
crypto map Testtag
!
interface Ethernet1
ip address 3.3.3.1 255.255.255.0
```

IKE Aggressive Mode Initiation RADIUS User Profile Example

The following is an example of a user profile on a RADIUS server that supports the Tunnel-Client-Endpoint and Tunnel-Password attributes:

```
user@cisco.com Password = "cisco", Service-Type = Outbound
Tunnel-Medium-Type = :1:IP,
Tunnel-Type = :1:ESP,
Cisco:Avpair = "ipsec:tunnel-password=cisco123",
Cisco:Avpair = "ipsec:key-exchange=ike"
```

Call Admission Control for IKE Configuration Examples

The following examples show how to configure Call Admission Control (CAC) for IKE:

- IKE Security Association Limit Configuration Example, page 24-26
- System Resource Limit Configuration Example, page 24-27

IKE Security Association Limit Configuration Example

The following example shows how to specify that there can be a maximum of 25 SAs before IKE starts rejecting new SA requests:

```
Router(config)# crypto call admission limit ike sa 25
```

System Resource Limit Configuration Example

The following example shows how to specify that IKE should drop SA requests when 90 percent of system resources are being used:

```
Router(config)# call admission limit 90
```

Dead Peer Detection Configuration Examples

The following examples show how to configure Dead Peer Detection (DPD):

- On-Demand DPD Configuration Example, page 24-27
- Periodic DPD Configuration Example, page 24-27
- DPD and Cisco IOS Keepalives with Multiple Peers in a Crypto Map Configuration Example, page 24-27

On-Demand DPD Configuration Example

The following example shows how to configure on-demand DPD messages. In this example, DPD messages will be sent every 60 seconds and every 5 seconds between retries if the peer does not respond:

```
Router(config)# crypto isakmp keepalive 60 5
```

Periodic DPD Configuration Example

The following example shows how to configure periodic DPD messages. In this example, DPD messages are to be sent at intervals of 10 seconds:

```
Router(config)# crypto isakmp keepalive 10 periodic
```

DPD and Cisco IOS Keepalives with Multiple Peers in a Crypto Map Configuration Example

The following example shows that DPD and Cisco IOS keepalives are used in conjunction with multiple peers in a crypto map configuration when IKE will be used to establish the security associations (SAs). In this example, an SA could be established to the IPSec peer at 10.0.0.1, 10.0.0.2, or 10.0.0.3.

```
Router(config)# crypto map green 1 ipsec-isakmp
Router(config-crypto-map)# set peer 10.0.0.1
Router(config-crypto-map)# set peer 10.0.0.2
Router(config-crypto-map)# set peer 10.0.0.3
Router(config-crypto-map)# set transform-set txfm
Router(config-crypto-map)# match address 101
```

IPSec NAT Transparency Configuration Example

The following example shows how to enable NAT keepalives to be sent every 20 seconds:

```
crypto isakmp policy 1
authentication pre-share
crypto isakmp key 1234 address 56.0.0.1
crypto isakmp nat keepalive 20
!
crypto ipsec transform-set t2 esp-des esp-sha-hmac
!
crypto map test2 10 ipsec-isakmp
set peer 56.0.0.1
set transform-set t2
match address 101
```

CHAPTER 25

Configuring Enhanced IPSec Features Using the IPSec VPN SPA

This chapter provides information about configuring enhanced IPSec features using the IPSec VPN SPA on the Cisco 7600 series router. It includes the following sections:

Note For detailed information on Cisco IOS IPSec cryptographic operations and policies, refer to the *Cisco IOS Security Configuration Guide* and *Cisco IOS Security Command Reference*.

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Tip To ensure a successful configuration of your VPN using the IPSec VPN SPA, read all of the configuration summaries and guidelines before you perform any configuration tasks.

Overview of Enhanced IPSec Features

IPSec is a framework of open standards developed by the Internet Engineering Task Force (IETF). It provides security for transmission of sensitive information over unprotected networks such as the Internet. IPSec acts at the network layer, protecting and authenticating IP packets between participating IPSec devices ("peers"), such as Cisco routers.

This chapter describes the advanced IPSec features that can be used to improve scalability and performance of your IPSec VPN.

Configuring Advanced Encryption Standard in a Transform Set

The Advanced Encryption Standard (AES) is a privacy transform for IPSec and Internet Key Exchange (IKE) that has been developed to replace the Data Encryption Standard (DES). AES is designed to be more secure than DES. AES offers a larger key size, while ensuring that the only known approach to decrypt a message is for an intruder to try every possible key. AES has a variable key length—the algorithm can specify a 128-bit key (the default), a 192-bit key, or a 256-bit key.

To configure the AES encryption algorithm within a transform set, use the **crypto ipsec transform-set** command beginning in global configuration mode as follows:

Router(config)# **crypto ipsec transform-set** *transform-set-name transform1*[*transform2*[*transform3*]]

...

Router(config-crypto-tran)# **exit**

In this command:

- *transform-set-name* specifies the name of the transform set.
- *transform1*[*transform2*[*transform3*]] defines IPSec security protocols and algorithms. To configure AES, you must choose from the following AES Encapsulating Security Payload (ESP) encryption transforms:
  - **esp-aes** specifies ESP with the 128-bit Advanced Encryption Standard (AES) encryption algorithm.
  - **esp-aes 192** specifies ESP with the 192-bit AES encryption algorithm.
  - **esp-aes 256** specifies ESP with the 256-bit AES encryption algorithm.

For other accepted *transformx* values, and more details on configuring transform sets, refer to the *Cisco IOS Security Command Reference*

Verifying the AES Transform Set

To verify the configuration of the transform set, enter the **show crypto ipsec transform-set** command:

```
Router# show crypto ipsec transform-set

Transform set transform-1:{esp-256-aes esp-md5-hmac}
will negotiate = {Tunnel, }
```

For more complete configuration information about AES support, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ft_aes.htm

For an AES configuration example, see the "Advanced Encryption Standard Configuration Example" section on page 25-22.

Configuring Look-Ahead Fragmentation

When a packet is nearly the size of the maximum transmission unit (MTU) of the outbound link of the encrypting router, and it is encapsulated with IPSec headers, it is likely to exceed the MTU of the outbound link. This causes packet fragmentation after encryption, which makes the decrypting router reassemble in the process path. Pre-fragmentation for IPSec VPNs increases the decrypting router's performance by enabling it to operate in the high performance CEF path instead of the process path.

The Look-Ahead Fragmentation (LAF) feature, also known as the Pre-Fragmentation for IPSEc VPNs feature, allows an encrypting router to predetermine the encapsulated packet size from information available in transform sets, which are configured as part of the IPSec security association (SA). If it is predetermined that the packet will exceed the MTU of the output interface, the packet is fragmented before encryption. This function avoids process-level reassembly before decryption and helps improve decryption performance and overall IPSec traffic throughput.

Look-Ahead Fragmentation Configuration Guidelines

Follow these guidelines when configuring Look-Ahead Fragmentation (LAF):

- Large packets can increase the IPSec packet size beyond the MTU causing the IPSec packets to be fragmented. When this situation occurs, the receiving IPSec peer must reassemble the packets prior to decryption. This action can cause serious loading for many VPN gateway devices. The solution is to fragment the packets before IPSec decryption and let the end devices bear the reassembly load.
- If there is no large packet connectivity through an IPSec peer, turn off LAF (the peer may be discarding fragments found inside the IPSec packets).
- If an IPSec peer is experiencing high CPU utilization with large packet flows, verify that LAF is enabled (the peer may be reassembling large packets).
- Look-Ahead Fragmentation for IPSec VPNs operates in IPSec tunnel mode and IPSec tunnel mode with GRE, but not with IPSec transport mode.
- Look-Ahead Fragmentation for IPSec VPNs configured on the decrypting router in a unidirectional traffic scenario does not improve the performance or change the behavior of either of the peers.
- Look-Ahead Fragmentation for IPSec VPNs functionality depends on the egress interface **crypto ipsec df-bit** configuration and the incoming packet "do not fragment" (DF) bit state. Refer to the following URL for details:

  http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080115533.html

Configuring Look-Ahead Fragmentation at the Interface

Look-Ahead Fragmentation is globally enabled by default. To enable or disable Look-Ahead Fragmentation for IPSec VPNs at the interface level, perform the following steps beginning in interface configuration mode:

Note Manually enabling or disabling this feature will override the global configuration.

	Command	Purpose
Step 1	Router(config-if)# **crypto ipsec fragmentation before-encryption**	Enables pre-fragmentation for IPSec VPNs on the interface.
Step 2	Router(config-if)# **crypto ipsec fragmentation after-encryption**	Disables pre-fragmentation for IPSec VPNs on the interface.

Configuring Look-Ahead Fragmentation Globally

Look-Ahead Fragmentation is globally enabled by default. To enable or disable pre-fragmentation for IPSec VPNs at the global level, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto ipsec fragmentation before-encryption**	Enables pre-fragmentation for IPSec VPNs globally.
Step 2	Router(config)# **crypto ipsec fragmentation after-encryption**	Disables pre-fragmentation for IPSec VPNs globally.

Verifying the Look-Ahead Fragmentation Configuration

To verify that Look-Ahead Fragmentation is enabled, consult the interface statistics on the encrypting router and the decrypting router. If fragmentation occurs on the encrypting router, and no reassembly occurs on the decrypting router, fragmentation is happening before encryption, and thus the packets are not being reassembled before decryption. This means that the feature is enabled.

Note This method of verification does not apply to packets destined for the decrypting router.

The first step to verifying that the feature is enabled is to enter the **show running-configuration** command on the encrypting router. If the feature is enabled, you will observe output similar to the following:

```
Router# show running-configuration

crypto isakmp policy 10
authentication pre-share
crypto isakmp key abcd123 address 25.0.0.7
crypto ipsec transform-set fooprime esp-3des esp-sha-hmac
crypto map bar 10 ipsec-isakmp
set peer 25.0.0.7
set transform-set fooprime
match address 102
```

If the feature has been disabled, you will observe output similar to the following:

```
Router# show running-configuration

crypto isakmp policy 10
authentication pre-share
crypto isakmp key abcd123 address 25.0.0.7
crypto ipsec transform-set fooprime esp-3des esp-sha-hmac
crypto ipsec fragmentation after-encryption
crypto map bar 10 ipsec-isakmp
set peer 25.0.0.7
set transform-set fooprime
match address 102
```

Next, enter the **show running-configuration interface** command to display statistics for the encrypting router egress interface. If the feature is enabled, you will observe output similar to the following:

```
Router# show running-configuration interface gigabitethernet 5/0/1

interface GigabitEthernet5/0/1

ip address 25.0.0.6 255.0.0.0
no ip mroute-cache
load-interval 30
duplex full
speed 100
crypto map bar
```

If the feature has been disabled, you will observe output similar to the following:

```
Router# show running-configuration interface gigabitethernet 5/0/1

interface GigabitEthernet5/0/1

ip address 25.0.0.6 255.0.0.0
no ip mroute-cache
load-interval 30
duplex full
speed 100
crypto map bar
crypto ipsec fragmentation after-encryption
```

For complete configuration information for LAF, refer to this URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080115533.html.

Configuring Reverse Route Injection (RRI)

Reverse Route Injection (RRI) provides the ability for static routes to be automatically inserted into the routing process for those networks and hosts protected by a remote tunnel endpoint. These protected hosts and networks are known as remote proxy identities.

Note RRI is only supported as of Cisco IOS Release 12.2(33)SRA.

Each route is created on the basis of the remote proxy network and mask, with the next hop to this network being the remote tunnel endpoint. By using the remote Virtual Private Network (VPN) router as the next hop, the traffic is forced through the crypto process to be encrypted.

After the static route is created on the VPN router, this information is propagated to upstream devices, allowing them to determine the appropriate VPN router to which to send returning traffic in order to maintain IPSec state flows. Being able to determine the appropriate VPN router is particularly useful if multiple VPN routers are used at a site to provide load balancing or failover or if the remote VPN devices are not accessible via a default route. Routes are created in either the global routing table or the appropriate virtual routing and forwarding (VRF) table.

RRI is applied on a per-crypto map basis, whether this is via a static crypto map or a dynamic crypto map template. For both dynamic and static maps, routes are created only at the time of IPSec SA creation. Routes are removed when the SAs are deleted. The **static** keyword can be added to the **reverse-route** command if the default behavior on static crypto maps is required; that is, if routes are created on the basis of the content of the crypto ACLs that are attached to the static crypto map.

RRI Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring RRI:

- IP routing should be enabled and static routes should be redistributed if dynamic routing protocols are to be used to propagate RRI-generated static routes.
- You can specify an interface or address as the explicit next hop to the remote VPN device. This functionality allows the overriding of a default route to properly direct outgoing encrypted packets.
- You can add a route tag value to any routes that are created using RRI. This route tag allows redistribution of groups of routes using route maps, allowing you to be selective about which routes enter your global routing table.
- RRI can be configured on the same crypto map that is applied to multiple router interfaces.
- The **remote-peer** [**static**] keyword creates two routes. One route is the standard remote proxy ID and the next hop is the remote VPN client tunnel address. The second route is the actual route to that remote tunnel endpoint and is used when a recursive lookup requires that the remote endpoint be reachable via "next hop." Creation of the second route for the actual next hop is important in the VRF case in which a default route must be overridden by a more explicit route.

  To reduce the number of routes created and support some platforms that do not readily facilitate route recursion, the **remote-peer** {*ip-address*} [**static**] keyword can be used to create one route only.
- For virtual IPSec interfaces, the reverse route option will create routes that list the virtual access interface as the next hop.
- For devices using an IPSec VPN SPA, reverse route specifies the next hop to be the interface, subinterface, or virtual LAN (VLAN) with the crypto map applied to it.

Configuring RRI Under a Static Crypto Map

To configure RRI under a static crypto map, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto map** {*map-name*} {*seq-name*} **ipsec-isakmp**	Creates or modifies a crypto map entry and enters crypto map configuration mode. • *map-name*—Name that identifies the map set. • *seq-num*—Sequence number assigned to the crypto map entry. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec SAs for protecting the traffic specified by this crypto map entry.
Step 2	Router(config-crypto-map)# **reverse-route** [[**static**] | **tag** {*tag-id*} [**static**] | **remote-peer** [**static**] | **remote-peer** {*ip-address*} [**static**]]	Creates source proxy information for a crypto map entry. • **static**—(Optional) Creates routes according to the existence of crypto access control lists (ACLs). • **tag** {*tag-id*}—Tag value that can be used as a "match" value for controlling redistribution via route maps. • **remote-peer** [**static**]—Two routes are created, one for the remote endpoint and one for route recursion to the remote endpoint via the interface to which the crypto map is applied. The **static** keyword is optional. • **remote-peer** {*ip-address*} [**static**]—One route is created to a remote proxy by way of a user-defined next hop. This next hop can be used to override a default route. The *ip-address* argument is required. The **static** keyword is optional.

Configuring RRI Under a Dynamic Crypto Map

To configure RRI under a dynamic crypto map, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto dynamic-map** {*dynamic-map-name*} {*dynamic-seq-name*}	Creates a dynamic crypto map entry and enters crypto map configuration mode. • *dynamic-map-name*—Name that identifies the map set. • *dynamic-seq-num*—Sequence number assigned to the crypto map entry.
Step 2	Router(config-crypto-map)# **reverse-route** [**tag** {*tag-id*} | **remote-peer** | **remote-peer** {*ip-address*}]	Creates source proxy information for a crypto map entry. • **tag** {*tag-id*}—Tag value that can be used as a "match" value for controlling redistribution via route maps. • **remote-peer**—Two routes are created, one for the remote endpoint and one for route recursion to the remote endpoint via the interface to which the crypto map is applied. • **remote-pee**r {*ip-address*}—One route is created to a remote proxy by way of a user-defined next hop. This next hop can be used to override a default route. The *ip-address* argument is required.

For more complete configuration information for RRI, refer to the following URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a00803854c7.html

For RRI configuration examples, see the "Reverse Route Injection Configuration Examples" section on page 25-22.

Configuring the IPSec Anti-Replay Window Size

Cisco IPSec authentication provides anti-replay protection against an attacker duplicating encrypted packets by assigning a unique sequence number to each encrypted packet. (Security association (SA) anti-replay is a security service in which the receiver can reject old or duplicate packets to protect itself against replay attacks.) The decryptor checks off the sequence numbers that it has seen before. The encryptor assigns sequence numbers in an increasing order. The decryptor remembers the value (X)of the highest sequence number that it has already seen. N is the window size of the decryptor. Any packet with a sequence number less than X — N is discarded. Currently, N is set at 64.

Note The IPSec Anti-Replay Window Size feature is only supported as of Cisco IOS Release 12.2(18)SXF6 and Cisco IOS Release 12.2(33)SRA.

At times, the 64-packet window size is not sufficient. For example, Cisco quality of service (QoS) gives priority to high-priority packets, which could cause some low-priority packets to be discarded even though they are not replayed packets. The IPSec Anti-Replay Window Size feature allows you to expand the window size so that sequence number information can be kept for more than 64 packets.

Expanding the IPSec Anti-Replay Window Size Globally

To expand the IPSec anti-replay window globally (so that it affects all SAs that are created—except for those that are specifically overridden on a per-crypto map basis), enter the **crypto ipsec security-association replay window size** command as follows beginning in global configuration mode:

Router(config)# **crypto ipsec security-association replay window size**[*n*]

In this command,*n* specifies the size of the window. The value can be 64, 128, 256, 512, or 1024. This value becomes the default value.

Expanding the IPSec Anti-Replay Window at the Crypto Map Level

To expand the IPSec anti-replay window on a crypto map basis (so that it affects those SAs that have been created using a specific crypto map or profile), perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto map** *map-name seq-num* **ipsec-isakmp**	Enters crypto map configuration mode and creates a crypto profile that provides a template for configuration of dynamically created crypto maps. • *map-name*—Name that identifies the map set. • *seq-num*—Sequence number assigned to the crypto map entry. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec SAs for protecting the traffic specified by this crypto map entry.
Step 2	Router(config-crypto-map)# **crypto ipsec security-association replay window size**[*n*]	Controls the SAs that are created using the policy specified by a particular crypto map, dynamic crypto map, or crypto profile. • *n*—(Optional) Size of the window. Values can be 64, 128,256, 512, or 1024. This value becomes the default value.

Verifying the IPSec Anti-Replay Window Size Configuration at the Crypto Map Level

To verify that IPSec anti-replay window size is enabled at a crypto map, enter the **show crypto map** command for that particular map. If anti-replay window size is enabled, the display will indicate that it is enabled and indicate the configured window size. If anti-replay window size is disabled, the results will indicate that also.

For more complete configuration information for IPSec anti-replay window size, refer to the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_14/gt_iarwe.htm

For IPSec anti-replay window size configuration examples, see the "IPSec Anti-Replay Window Size Configuration Examples" section on page 25-23.

Configuring an IPSec Preferred Peer

The IP Security (IPSec) Preferred Peer feature allows you to control the circumstances by which multiple peers on a crypto map are tried in a failover scenario. If there is a default peer, the next time a connection is initiated, the connection is directed to the default peer instead of to the next peer in the peer list. If all connections to the current peer time out, the next time a connection is initiated, it is directed to the default peer.

Note The IPSec Preferred Peer feature is only supported as of Cisco IOS Release 12.2(33)SRA.

This feature includes the following capabilities:

- Default peer configuration

  If a connection timeout occurs, the connection to the current peer is closed. The **set peer** command allows you to configure the first peer as the default peer. If there is a default peer, the next time a connection is initiated, the connection is directed to the default peer instead of to the next peer in the peer list. If the default peer is unresponsive, the next peer in the peer list becomes the current peer and future connections through the crypto map try that peer.

  This capability is useful when traffic on a physical link stops due to the failure of a remote peer. DPD indicates that the remote peer is unavailable, but that peer remains the current peer.

  A default peer facilitates the failover to a preferred peer that was previously unavailable, but has returned to service. Users can give preference to certain peers in the event of a failover. This is useful if the original failure was due to a network connectivity problem rather than failure of the remote peer.

  To configure a default peer, see the "Configuring a Default Peer" section on page 25-13.

- IPSec idle timer with default peer configuration

  When a router running Cisco IOS software creates an IPSec security association (SA) for a peer, resources must be allocated to maintain the SA. The SA requires both memory and several managed timers. For idle peers, these resources are wasted. If enough resources are wasted by idle peers, the router could be prevented from creating new SAs with other peers.

  IPSec SA idle timers increase the availability of resources by deleting SAs associated with idle peers. Because IPSec SA idle timers prevent the wasting of resources by idle peers, more resources are available to create new SAs when required. (If IPSec SA idle timers are not configured, only the global lifetimes for IPSec SAs are applied. SAs are maintained until the global timers expire, regardless of peer activity.)

  When both an IPSec SA idle timer and a default peer are configured and all connections to the current peer time out, the next time a connection is initiated it is directed to the default peer configured in the **set peer** command. If a default peer is not configured and there is a connection timeout, the current peer remains the one that timed out.

  This enhancement helps facilitate a failover to a preferred peer that was previously unavailable but is in service now.

  To configure an IPSec idle timer, see the "Configuring the IPSec Idle Timer with a Default Peer" section on page 25-14.

IPSec Preferred Peer Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring an IPSec preferred peer:

- Follow these guidelines and restrictions when configuring a default peer:
  - The default peer feature must be used in conjunction with Dead Peer Detection (DPD). It is most effective on a remote site running DPD in periodic mode. DPD detects the failure of a device quickly and resets the peer list so that the default peer is tried for the next attempted connection.
  - Only one peer can be designated as the default peer in a crypto map.
  - The default peer must be the first peer in the peer list.
- Follow these guidelines and restrictions when configuring IPSec idle timer usage with a default peer:
  - The IPSec idle timer usage with a default peer feature works only on the crypto map for which it is configured. You cannot configure the capability globally for all crypto maps.
  - If there is a global idle timer, the crypto map idle timer value must be different from the global value; otherwise the idle timer is not added to the crypto map.

Configuring a Default Peer

To configure a default peer, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto map** *map-name seq-num* [**ipsec-isakmp**] [**dynamic** *dynamic-map-name*] [**discover**] [**profile** *profile-name*]	Enters crypto map configuration mode and creates a crypto profile that provides a template for configuration of dynamically created crypto maps. • *map-name*—Name that identifies the map set. • *seq-num*—Sequence number assigned to the crypto map entry. • **ipsec-isakmp**—(Optional)Indicates that IKE will be used to establish the IPSec SAs for protecting the traffic specified by this crypto map entry. • **dynamic** *dynamic-map-name*—(Optional) Specifies the name of the dynamic crypto map set that should be used as the policy template. • **discover**—(Optional) Enables peer discovery. By default, peer discovery is not enabled. • **profile** *profile-name*—(Optional) Name of the crypto profile being created.
Step 2	Router(config-crypto-map)# **set peer** {*host-name* [**dynamic**] [**default**] | *ip-address* [**default**]}	Specifies an IPSec peer in a crypto map entry. Ensures that the first peer specified is defined as the default peer. • *host-name*—Specifies the IPSec peer by its host name. This is the peer's host name concatenated with its domain name (for example, myhost.example.com). • **dynamic**—(Optional) The host name of the IPSec peer will be resolved via a domain name server (DNS) lookup right before the router establishes the IPSec tunnel. • **default**—(Optional) If there are multiple IPSec peers, designates that the first peer is the default peer. • *ip-address*—Specifies the IPSec peer by its IP address.
Step 3	Router(config-crypto-map)# **exit**	Exits crypto map configuration mode and returns to global configuration mode.

Configuring the IPSec Idle Timer with a Default Peer

To configure the IPSec idle timer with a default peer, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto map** *map-name seq-num* [**ipsec-isakmp**] [**dynamic** *dynamic-map-name*] [**discover**] [**profile** *profile-name*]	Enters crypto map configuration mode and creates a crypto profile that provides a template for configuration of dynamically created crypto maps. • *map-name*—Name that identifies the map set. • *seq-num*—Sequence number assigned to the crypto map entry. • **ipsec-isakmp**—(Optional) Indicates that IKE will be used to establish the IPSec SAs for protecting the traffic specified by this crypto map entry. • **dynamic** *dynamic-map-name*—(Optional) Specifies the name of the dynamic crypto map set that should be used as the policy template. • **discover**—(Optional) Enables peer discovery. By default, peer discovery is not enabled. • **profile** *profile-name*—(Optional) Name of the crypto profile being created.
Step 2	Router(config-crypto-map)# **set security-association idle-time** *seconds* [**default**]	Specifies the maximum amount of time for which the current peer can be idle before the default peer is used. • *seconds*—Number of seconds for which the current peer can be idle before the default peer is used. Valid values are 60 to 86400. • **default**—(Optional) Specifies that the next connection is directed to the default peer.
Step 3	Router(config-crypto-map)# **exit**	Exits crypto map configuration mode and returns to global configuration mode.

For complete configuration information for IPSec preferred peer, refer to this URL:

http://lbj.cisco.com/push_targets1/ucdit/cc/td/doc/product/software/ios123/123newft/123t/123t_14/gt_ipspp.htm

For IPSec preferred peer configuration examples, see the "IPSec Preferred Peer Configuration Examples" section on page 25-26.

Configuring IPSec Security Association Idle Timers

When a router running Cisco IOS software creates an IPSec SA for a peer, resources must be allocated to maintain the SA. The SA requires both memory and several managed timers. For idle peers, these resources are wasted. If enough resources are wasted by idle peers, the router could be prevented from creating new SAs with other peers. The IPSec Security Association Idle Timers feature provides a configurable idle timer to monitor SAs for activity, allowing SAs for idle peers to be deleted. The idle timers can be configured either globally, on a per-crypto map basis, or through an ISAKMP profile. The benefits of this feature include the following:

- Increased availability of resources
- Improved scalability of Cisco IOS IPSec deployments

IPSec Security Association Idle Timer Configuration Guidelines

Follow these guidelines when configuring idle timers on a per-crypto map basis:

- The IPSec VPN SPA rounds up the CLI-configured interval to the nearest 10-minute interval. For example, if you configured 12 minutes for idle timeout, the IPSec VPN SPA uses a value of 20 minutes for idle timeout.
- Because of the way the IPSec VPN SPA does idle timeout detection, it can take anywhere between one to three (ten-minute) intervals for idle timeout detection. For example, if you configured 12 minutes for idle timeout, idle timeout could happen anywhere between 20 to 60 minutes.
- When the idle timer is configured globally, the idle timer configuration is applied to all SAs.
- When the idle timer is configured for a crypto map, the idle timer configuration is applied to all SAs under the specified crypto map.

Configuring the IPSec SA Idle Timer Globally

To configure the IPSec SA idle timer globally, enter the **crypto ipsec security-association idle-time** command in global configuration mode as follows:

Router(config)# **crypto ipsec security-association idle-time** *seconds*

In this command, *seconds* specifies the time, in seconds, that the idle timer will allow an inactive peer to maintain an SA. Valid values range from 60 to 86400.

Configuring the IPSec SA Idle Timer per Crypto Map

To configure the IPSec SA idle timer for a specified crypto map, use the **set security-association idle-time** command within a crypto map configuration beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto map** *map-name seq-number* **ipsec-isakmp**	Creates or modifies a crypto map entry and enters crypto map configuration mode. • *map-name*—Name that identifies the crypto map set. • *seq-number*—Sequence number you assign to the crypto map entry. Lower values have higher priority. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec security associations.
Step 2	Router(config-crypto-map)# **set security-association idle-time** *seconds*	Configures the IPSec SA idle timer. • *seconds*—Time, in seconds, that the idle timer will allow an inactive peer to maintain an SA. Valid values range from 60 to 86400.

For detailed information on configuring IPSec SA idle timers, refer to the following Cisco IOS documentation:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t15/ftsaidle.htm#wp1027129

For IPSec SA idle timer configuration examples, see the "IPSec Security Association Idle Timer Configuration Examples" section on page 25-26.

Configuring Crypto Map-Based Distinguished Name Filtering

The Distinguished Name-Based Crypto Maps feature allows you to configure the router to restrict access to selected encrypted interfaces for those peers with specific certificates, especially certificates with particular Distinguished Names (DNs).

Previously, if the router accepted a certificate or a shared secret from the encrypting peer, Cisco IOS did not have a method of preventing the peer from communicating with any encrypted interface other than the restrictions on the IP address of the encrypting peer. This feature allows you to configure which crypto maps are usable to a peer based on the DN that a peer used to authenticate itself, thereby enabling you to control which encrypted interfaces a peer with a specified DN can access. You can configure a DN-based crypto map that can be used only by peers that have been authenticated by a DN or one that can be used only by peers that have been authenticated by a hostname.

Crypto Map-Based Distinguished Name Filtering Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring crypto map-based DN filtering:

- If you restrict access to a large number of DNs, it is recommended that you specify a few number of crypto maps referring to large identity sections instead of specifying a large number of crypto maps referring to small identity sections.

To configure a crypto map-based DN that can be used only by peers that have been authenticated by a DN, or one that can be used only by peers that have been authenticated by a host name, perform the following steps beginning in global configuration mode:

<table>
<tr><th></th><th>Command</th><th>Purpose</th></tr>
<tr><td>Step 1</td><td>Router(config)# crypto isakmp policy priority
...
Router(config-isakmp)# exit</td><td>Defines an ISAKMP policy and enters ISAKMP policy configuration mode.
• priority—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest.
Creates an ISAKMP policy at each peer.
For details on configuring an ISAKMP policy, see the Cisco IOS Security Configuration Guide.</td></tr>
<tr><td>Step 2</td><td>Router(config)# crypto map map-name seq-number ipsec-isakmp</td><td>Creates or modifies a crypto map entry and enters crypto map configuration mode.
• map-name—Name that identifies the crypto map set.
• seq-number—Sequence number you assign to the crypto map entry. Lower values have higher priority.
• ipsec-isakmp—Indicates that IKE will be used to establish the IPSec security associations.</td></tr>
<tr><td>Step 3</td><td>Router(config-crypto-map)# identity name
...
Router(config-crypto-map)# exit</td><td>Applies the identity to the crypto map.
• name—Identity of the router, which is associated with the given list of DNs.
When this command is applied, only the hosts that match a configuration listed within the identity name can use the specified crypto map.
Note If the identity command does not appear within the crypto map, the encrypted connection does not have any restrictions other than the IP address of the encrypting peer.
Specify any other policy values appropriate to your configuration.
For details on configuring a crypto map, see the Cisco IOS Security Configuration Guide.</td></tr>
</table>

	Command	Purpose
Step 4	Router(config)# **crypto identity** *name*	Configures the identity of a router with the given list of DNs in the certificate of the router and enters crypto identity configuration mode. • *name*—Enter the name value specified in Step 3.
Step 5	Router(crypto-identity)# **dn** *name*=**string** [,*name*=**string**] or Router(crypto-identity) # **fqdn** *name*	Associates the identity of the router with either a DN or host name (FQDN) to restrict access to peers with specific certificates. • *name*=**string**—Enter the DN in the certificate of the router. Optionally, you can associate more than one DN. • **fqdn** *name*—Enter the host name that the peer used to authenticate itself (FQDN) or the DN in the certificate of the router. The identity of the peer must match the identity in the exchanged certificate.

For complete configuration information about Distinguished Name-Based Crypto Maps, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t4/ftdnacl.htm

For a crypto map-based distinguished name filtering configuration example, see the "Crypto Map-Based Distinguished Name Filtering Configuration Example" section on page 25-27.

Configuring MTU Settings

The Cisco IOS software supports several types of configurable maximum transmission unit (MTU) options at various levels of the protocol stack. You should ensure that all MTU values are consistent to avoid unneccessary fragmentation of packets.

MTU Settings Configuration Guidelines and Restrictions

Follow these guidelines and note these restrictions when configuring MTU settings for an IPSec VPN SPA:

- As a general rule, do not change the MTU value unless you have a specific need to do so.
- The MTU value used by the IPSec VPN SPA for fragmentation decisions is based on the MTU value of the secure port as follows:
  - Routed ports—Use the MTU value of their associated secure port.
  - Access ports—Use the MTU value of the secure port associated with their interface VLAN.
  - Trunk ports—Use the MTU value of the secure port associated with their interface VLAN.
- If you have GRE tunneling configured, the "ip MTU" of the GRE tunnel is used.

Note For additional information on fragmentation of packets, see the "Configuring Look-Ahead Fragmentation" section on page 25-3.

To change the MTU value on a Gigabit Ethernet interface, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface gigabitethernet** *slot/subslot/port*	Enters interface configuration mode for the Gigabit Ethernet interface. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 2	Router(config-if)# **mtu** *bytes*	Configures the maximum transmission unit (MTU) size for the interface. • *bytes*—Valid range is 1500 to 9216. The default values used are described as above.
Step 3	Router(config-if)# **end**	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying the MTU Size

To verify the MTU sizes for an interface, enter the **show interfaces** commands.

For example, to display the MTU value for a secure port, enter the following command:

```
Router# show interfaces g1/1/1

GigabitEthernet1/1/1 is up, line protocol is up (connected)
Hardware is C6k 1000Mb 802.3, address is 000a.8ad8.1c4a (bia 000a.8ad8.1c4a)
MTU 9216 bytes, BW 1000000 Kbit, DLY 10 usec,
reliability 255/255, txload 1/255, rxload 1/255
.
.
.
```

To display the MTU value for a GRE tunnel, enter the following command:

```
Router# show ip interfaces tunnel 2

Tunnel2 is up, line protocol is up
Internet address is 11.1.0.2/16
Broadcast address is 255.255.255.255
Address determined by non-volatile memory
MTU is 1450 bytes
.
.
.
```

Configuring Sequenced ACLs

Access control lists (ACLs) are made up of access control entries (ACEs). With sequenced ACLs, ACEs can be entered with a sequence number in front of the ACE and the ACEs are then processed by sequence number. Additionally, ACEs can be deleted one at a time by using the sequence number in the front of the ACE that you want to delete. The sequence numbers do not appear in the configuration but they can be displayed using the **show access-list** command.

Note If an ACE is removed or modified, the ACL is reconfigured on the SPA. This might result in tearing down existing sessions.

Configuring Deny Policy Enhancements for ACLs

Specifying a deny address range in an ACL results in "jump" behavior. When a denied address range is hit, it forces the search to "jump" to the beginning of the ACL associated with the next sequence in a crypto map and continue the search. If you want to pass clear traffic on these addresses, you must insert a deny address range for each sequence in a crypto map. In turn, each permit list of addresses inherits all the deny address ranges specified in the ACL. A deny address range causes the software to do a subtraction of the deny address range from a permit list, and creates multiple permit address ranges that need to be programmed in hardware. This behavior can cause repeated address ranges to be programmed in the hardware for a single deny address range, resulting in multiple permit address ranges in a single ACL. To avoid this problem, use the **crypto ipsec ipv4 deny-policy** {**jump** | **clear** | **drop**} command set as follows:

- The **jump** keyword results in the standard "jump" behavior.
- The **clear** keyword allows a deny address range to be programmed in hardware, the deny addresses are then filtered out for encryption and decryption. When a deny address is hit, the search is stopped and traffic is allowed to pass in the clear (unencrypted) state.
- The **drop** keyword causes traffic to be dropped when a deny address is hit.

Note that the **clear** and **drop** keywords can be used to prevent repeated address ranges from being programmed in the hardware, resulting in more efficient TCAM space utilization.

Deny Policy Enhancements for ACLs Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring the deny policy enhancements:

- The **crypto ipsec ipv4 deny-policy** {**jump** | **clear** | **drop**} command is a global command that is applied to a single IPSec VPN SPA. The specified keyword (**jump**, **clear**, or **drop**) is propagated to the ACE software of the IPSec VPN SPA. The default behavior is **jump**.
- If you apply the specified keyword (**jump**, **clear**, or **drop**) when crypto maps are already configured on the IPSec VPN SPA, all existing IPSec sessions are temporarily removed and restarted, which impacts traffic on your network.
- The number of deny entries that can be specified in an ACL are dependent on the keyword specified:
  - **jump**—Supports up to 8 deny entries in an ACL
  - **clear**—Supports up to 1000 deny entries in an ACL
  - **drop**—Supports up to 1000 deny entries in an ACL

For a deny policy enhancements configuration example, see the "Deny Policy Enhancements for ACLs Configuration Example" section on page 25-28.

Configuring Priority Queueing Using the IPSec VPN SPA

The IPSec VPN SPA supports two-level, strict-priority QoS (high priority versus low priority). To take advantage of the IPSec VPN SPA's QoS capability, you must use standard QoS commands to ensure that the CoS of packets are marked on ingress. You must configure the CoS map for the IPSec VPN SPA inside and outside ports. The IPSec VPN SPA behaves according to the settings of the inside and outside ports. You must enable QoS globally for the IPSec VPN SPA to acknowledge the CoS mapping.

For example, if the CoS map of the inside and outside ports map CoS value 5 to the high-priority queue and you have globally enabled QoS, the IPSec VPN SPA will give traffic marked CoS 5 higher priority than traffic marked with any of the other seven CoS values. If you alter the CoS map of the inside and outside ports so that CoS 6 additionally maps to the high-priority queue, then packets marked with either CoS 5 or CoS 6 will be given higher priority within the IPSec VPN SPA.

As many as three high-priority CoS map values are supported per IPSec VPN SPA. When global QoS is enabled, the CoS value of 5 is preconfigured. This allows you to add only two more values in addition to the preconfigured CoS 5 value.

The IPSec VPN SPA uses the QoS capabilities of the Cisco 7600 series router software. Before configuring QoS for the IPSec VPN SPA, refer to this URL:

http://www.cisco.com/en/US/products/hw/switches/ps700/products_tech_note09186a008014a29f.shtml

For priority queueing in the IPSec VPN SPA configuration examples, see the "Priority Queueing Using the IPSec VPN SPA Configuration Examples" section on page 25-28.

Configuration Examples

This section provides examples of the following configurations:

Advanced Encryption Standard Configuration Example

The following example is sample output from the **show running-config** command. In this example, the Advanced Encryption Standard (AES) 256-bit key is enabled.

```
Router# show running-config

Current configuration : 1665 bytes
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname "Router1"
ip subnet-zero
no ip domain lookup
ip audit notify log
ip audit po max-events 100
crypto isakmp policy 10
encryption aes 256
authentication pre-share
lifetime 180
crypto isakmp key cisco123 address 10.0.110.1
crypto ipsec transform-set aesset esp-aes 256 esp-sha-hmac
mode transport
crypto map aesmap 10 ipsec-isakmp
set peer 10.0.110.1
set transform-set aesset
```

Reverse Route Injection Configuration Examples

The following examples show how to configure RRI:

- RRI Under a Static Crypto Map Configuration Example, page 25-22
- RRI Under a Dynamic Crypto Map Configuration Example, page 25-23
- RRI with Existing ACLs Configuration Example, page 25-23
- RRI for Two Routes Configuration Example, page 25-23
- RRI via a User-Defined Hop Configuration Example, page 25-23

RRI Under a Static Crypto Map Configuration Example

The following example shows how to configure RRI under a static crypto map. In this example, the RRI-created route has been tagged with a tag number. This tag number can then be used by a routing process to redistribute the tagged route via a route map:

```
Router(config)# crypto map mymap 1 ipsec-isakmp
Router(config-crypto-map)# reverse-route tag 5
```

RRI Under a Dynamic Crypto Map Configuration Example

The following example shows how to configure Reverse Route Injection under a dynamic crypto map:

```
Router(config)# crypto dynamic-map mymap 1
Router(config-crypto-map)# reverse-route remote peer 10.1.1.1
```

RRI with Existing ACLs Configuration Example

The following example shows how to configure RRI for a situation in which there are existing ACLs:

```
Router(config)# crypto map mymap 1 ipsec-isakmp
Router(config-crypto-map)# set peer 172.17.11.1
Router(config-crypto-map)# reverse-route static
Router(config-crypto-map)# set transform-set esp-3des-sha
Router(config-crypto-map)# match address 101

access-list 101 permit ip 192.168.1.0 0.0.0.255 172.17.11.0 0.0.0.255
```

Note In Cisco IOS Release 12.3(14)T and later, for the static map to retain the behavior of creating routes on the basis of crypto ACL content, the **static** keyword must be specified.

RRI for Two Routes Configuration Example

The following example shows how to configure two routes, one for the remote endpoint and one for route recursion to the remote endpoint via the interface on which the crypto map is configured:

```
Router(config-crypto-map)# reverse-route remote-peer
```

RRI via a User-Defined Hop Configuration Example

The following example shows that one route has been created to the remote proxy via a user-defined next hop. This next hop should not require a recursive route lookup unless it will recurse to a default route.

```
Router(config-crypto-map)# reverse-route remote-peer 10.4.4.4
```

IPSec Anti-Replay Window Size Configuration Examples

The following examples show how to configure the IPSec anti-replay window size:

- IPSec Anti-Replay Window Global Configuration Example, page 25-24
- IPSec Anti-Replay Window per Crypto Map Configuration Example, page 25-25

IPSec Anti-Replay Window Global Configuration Example

The following example shows that the anti-replay window size has been set globally to 1024:

```
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname VPN-Gateway1
!
boot-start-marker
boot-end-marker
!
clock timezone EST 0
no aaa new-model
ip subnet-zero
!
ip audit po max-events 100
no ftp-server write-enable
!
crypto isakmp policy 10
authentication pre-share
crypto isakmp key cisco123
address 192.165.201.2
!
crypto ipsec security-association replay window-size 1024
!
crypto ipsec transform-set basic esp-des esp-md5-hmac
!
crypto map mymap 10 ipsec-isakmp
set peer 192.165.201.2
set transform-set basic
match address 101
!
interface Ethernet0/0
ip address 192.168.1.1 255.255.255.0
!
interface Serial1/0
ip address 192.165.200.2 255.255.255.252
serial restart-delay 0
crypto map mymap
!
ip classless
ip route 0.0.0.0 0.0.0.0 192.165.200.1
no ip http server
no ip http secure-server
!
access-list 101 permit ip 192.168.1.0 0.0.0.255 172.16.2.0 0.0.0.255
!access-list 101 remark Crypto ACL
!
control-plane
!
line con 0
line aux 0
line vty 0 4
end
```

IPSec Anti-Replay Window per Crypto Map Configuration Example

The following example shows that anti-replay checking is disabled for IPSec connections to 172.150.150.2, but enabled (and the default window size is 64) for IPSec connections to 172.150.150.3 and 172.150.150.4:

```
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname dr_whoovie
!
enable secret 5 $1$KxKv$cbqKsZtQTLJLGPN.tErFZ1
enable password ww
!
ip subnet-zero
cns event-service server
crypto isakmp policy 1
authentication pre-share
crypto isakmp key cisco170
address 172.150.150.2
crypto isakmp key cisco180
address 172.150.150.3
crypto isakmp key cisco190
address 172.150.150.4
crypto ipsec transform-set 170cisco esp-des esp-md5-hmac
crypto ipsec transform-set 180cisco esp-des esp-md5-hmac
crypto ipsec transform-set 190cisco esp-des esp-md5-hmac
crypto map ETH0 17 ipsec-isakmp
set peer 172.150.150.2
set security-association replay disable
set transform-set 170cisco
match address 170
crypto map ETH0 18 ipsec-isakmp
set peer 150.150.150.3
set transform-set 180cisco
match address 180
crypto map ETH0 19 ipsec-isakmp
set peer 150.150.150.4
set transform-set 190cisco
match address 190
!
interface Ethernet0
ip address 172.150.150.1 255.255.255.0
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
no mop enabled
crypto map ETH0
!
interface Serial0
ip address 172.160.160.1 255.255.255.0
no ip directed-broadcast
no ip mroute-cache
no fair-queue
!
ip classless
ip route 172.170.170.0 255.255.255.0 172.150.150.2
ip route 172.180.180.0 255.255.255.0 172.150.150.3
ip route 172.190.190.0 255.255.255.0 172.150.150.4
no ip http server
!
access-list 170 permit ip 172.160.160.0 0.0.0.255 172.170.170.0 0.0.0.255
```

```
access-list 180 permit ip 172.160.160.0 0.0.0.255 172.180.180.0 0.0.0.255
access-list 190 permit ip 172.160.160.0 0.0.0.255 172.190.190.0 0.0.0.255
!
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
!
line con 0
transport input none
line aux 0
line vty 0 4
password ww
login
end
```

IPSec Preferred Peer Configuration Examples

The following examples show how to configure an IPSec preferred peer:

- Default Peer Configuration Example, page 25-26
- IPSec Idle Timer with Default Peer Configuration Example, page 25-26

Default Peer Configuration Example

The following example shows how to configure a default peer. In this example, the first peer, at IP address 1.1.1.1, is the default peer:

```
Router(config)# crypto map tohub 1 ipsec-isakmp
Router(config-crypto-map)# set peer 1.1.1.1 default
Router(config-crypto-map)# set peer 2.2.2.2
Router(config-crypto-map)# exit
```

IPSec Idle Timer with Default Peer Configuration Example

The following example shows how to configure an IPSec idle timer with a default peer. In the following example, if the current peer is idle for 120 seconds, the default peer 1.1.1.1 (which was specified in the **set peer** command) is used for the next attempted connection:

```
Router (config)# crypto map tohub 1 ipsec-isakmp
Router(config-crypto-map)# set peer 1.1.1.1 default
Router(config-crypto-map)# set peer 2.2.2.2
Router(config-crypto-map)# set security-association idle-time 120 default
Router(config-crypto-map)# exit
```

IPSec Security Association Idle Timer Configuration Examples

The following examples show how to configure the IPSec SA idle timer:

- IPSec SA Idle Timer Global Configuration Example, page 25-27
- IPSec SA Idle Timer per Crypto Map Configuration Example, page 25-27

IPSec SA Idle Timer Global Configuration Example

The following example globally configures the IPSec SA idle timer to drop SAs for inactive peers after 600 seconds:

```
Router(config)# crypto ipsec security-association idle-time 600
```

IPSec SA Idle Timer per Crypto Map Configuration Example

The following example configures the IPSec SA idle timer for the crypto map named “test” to drop SAs for inactive peers after 600 seconds:

```
Router(config) # crypto map test 1 ipsec-isakmp
Router(config-crypto-map)# set security-association idle-time 600
```

Crypto Map-Based Distinguished Name Filtering Configuration Example

The following example shows how to configure a crypto map-based distinguished name that has been authenticated by the DN and host name. Comments are included inline to explain various commands.

```
! DN based crypto maps require you to configure an IKE policy at each peer.
crypto isakmp policy 15
encryption 3des
hash md5
authentication rsa-sig
group 2
lifetime 5000
crypto isakmp policy 20
authentication pre-share
lifetime 10000
crypto isakmp key 1234567890 address 171.69.224.33
!
!The following is an IPSec crypto map (part of IPSec configuration). It can be used only
! by peers that have been authenticated by DN and if the certificate belongs to BigBiz.
crypto map map-to-bigbiz 10 ipsec-isakmp
set peer 172.21.114.196
set transform-set my-transformset
match address 124
identity to-bigbiz
!
crypto identity to-bigbiz
dn ou=BigBiz
!
! This crypto map can be used only by peers that have been authenticated by hostname
!and if the certificate belongs to little.com.
crypto map map-to-little-com 10 ipsec-isakmp
set peer 172.21.115.119
set transform-set my-transformset
match address 125
identity to-little-com
!
crypto identity to-little-com
fqdn little.com
!
```

Deny Policy Enhancements for ACLs Configuration Example

The following example shows a configuration using the deny-policy **clear** option. In this example, when a deny address is hit, the search will stop and traffic will be allowed to pass in the clear (unencrypted) state:

```
Router(config)# crypto ipsec ipv4 deny-policy clear
```

Priority Queueing Using the IPSec VPN SPA Configuration Examples

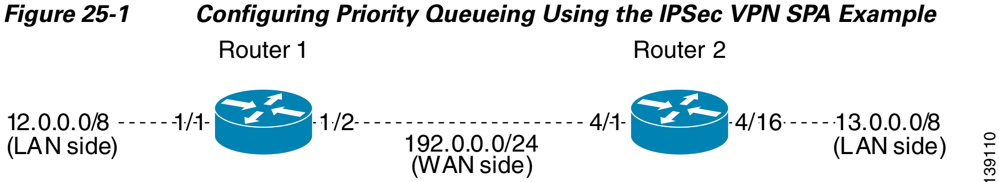
The following examples show how to configure priority queueing using the IPSec VPN SPA for the configuration shown in Figure 25-1 on page 25-28:

- Priority Queueing Using the IPSec VPN SPA (Router 1) Configuration Example, page 25-28
- Priority Queueing Using the IPSec VPN SPA (Router 2) Configuration Example, page 25-30

A summary of the router configuration that is used in the examples is as follows:

- The IPSec VPN SPA is in slot 3, subslot 0 on both routers.
- An IPSec tunnel that is between router 1 and router 2 encrypts all traffic.
- Both routers are configured so that IP packets with ToS 5 or ToS 7 go to high priority.
- To highlight the QoS configuration steps in the configuration examples, three exclamation points (!!!) precede each QoS-related command.

Figure 25-1 Configuring Priority Queueing Using the IPSec VPN SPA Example



Priority Queueing Using the IPSec VPN SPA (Router 1) Configuration Example

The following example shows the priority queueing using the IPSec VPN SPA configuration for router 1 as shown in Figure 25-1:

```
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service compress-config
!
hostname Switch 1
!
logging snmp-authfail
!
ip subnet-zero
!
no ip domain-lookup
ip host tftp 223.255.254.254
!
mpls ldp logging neighbor-changes
mls flow ip destination
```

```
mls flow ipx destination
!!! Enables qos globally
mls qos
!
crypto isakmp policy 10
 authentication pre-share
crypto isakmp key 12345 address 192.0.0.2
!
crypto ipsec transform-set 3des_sha1_ts esp-3des esp-sha-hmac
!
crypto map cmap2 100 ipsec-isakmp
 set peer 192.0.0.2
 set transform-set 3des_sha1_ts
 match address acl0
!
spanning-tree extend system-id
no spanning-tree vlan 2
!
redundancy
 mode rpr-plus
 main-cpu
  auto-sync running-config
  auto-sync standard
!
interface GigabitEthernet1/1
 ip address 12.0.0.1 255.0.0.0
 no keepalive
 speed nonegotiate
!!! Trust incoming ip precedence bits (from LAN side)
 mls qos trust ip-precedence
!
interface GigabitEthernet1/2
 no ip address
!!! Trust incoming ip precedence bits (from WAN side)
 mls qos trust ip-precedence
 crypto connect vlan 2
!
interface GigabitEthernet3/0/1
 no ip address
!!! COS 5 and 7 will go to high priority queue
 priority-queue cos-map 1 5 7
!!! Trust Ethernet frame COS bits
 mls qos trust cos
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2,1002-1005
 switchport mode trunk
 cdp enable
!
interface GigabitEthernet3/0/2
 no ip address
!!! This command is added automatically when /1 was configured
 priority-queue cos-map 1 5 7
!!! Trust Ethernet frame COS bits
 mls qos trust cos
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 cdp enable
```

```
 spanning-tree portfast trunk
!
interface Vlan2
 ip address 192.0.0.1 255.255.255.0
 no mop enabled
 crypto map cmap2
  crypto engine slot 3/0
!
ip classless
ip route 13.0.0.0 255.0.0.0 192.0.0.2
no ip http server
no ip http secure-server
!
ip access-list extended acl0
 permit ip any any
!
line con 0
 exec-timeout 0 0
line vty 0 4
 login
 transport input lat pad mop telnet rlogin udptn nasi ssh acercon
!
end
```

Priority Queueing Using the IPSec VPN SPA (Router 2) Configuration Example

The following example shows the priority queueing using the IPSec VPN SPA configuration for router 2 as shown in Figure 25-1:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Switch 2
!
logging snmp-authfail
no logging console
!
ip subnet-zero
!
no ip domain-lookup
ip host tftp 223.255.254.254
!
mpls ldp logging neighbor-changes
mls flow ip destination
mls flow ipx destination
!!! Enables qos globaly
mls qos
!
crypto isakmp policy 10
 authentication pre-share
crypto isakmp key 12345 address 192.0.0.1
!
crypto ipsec transform-set 3des_sha1_ts esp-3des esp-sha-hmac
!
crypto map cmap2 100 ipsec-isakmp
 set peer 192.0.0.1
 set transform-set 3des_sha1_ts
 match address acl0
!
no spanning-tree vlan 2
```

```
!
redundancy
 mode rpr-plus
 main-cpu
  auto-sync running-config
  auto-sync standard
!
interface GigabitEthernet3/0/1
 no ip address
!!! COS 5 and 7 will go to high priority queue
 priority-queue cos-map 1 5 7
!!! Trust Ethernet frame COS bits
 mls qos trust cos
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2,1002-1005
 switchport mode trunk
 cdp enable
!
interface GigabitEthernet3/0/2
 no ip address
!!! This command is added automatically when /1 was configured
 priority-queue cos-map 1 5 7
!!! Trust Ethernet frame COS bits
 mls qos trust cos
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 cdp enable
 spanning-tree portfast trunk
!
interface GigabitEthernet4/1
 no ip address
!!! Trust incoming ip precedence bits (from WAN side)
 mls qos trust ip-precedence
 crypto connect vlan 2
!
interface GigabitEthernet4/16
 ip address 13.0.0.1 255.0.0.0
!!! Trust incoming ip precedence bits (from LAN side)
 mls qos trust ip-precedence
!
interface Vlan2
 ip address 192.0.0.2 255.255.255.0
 no mop enabled
 crypto map cmap2
  crypto engine slot 3/0
!
ip classless
ip route 12.0.0.0 255.0.0.0 192.0.0.1
no ip http server
no ip http secure-server
ip pim bidir-enable
!
ip access-list extended acl0
 permit ip any any
!
arp 127.0.0.12 0000.2100.0000 ARPA
!
```

```
line con 0
 exec-timeout 0 0
line vty 0 4
 password a
 login
 transport input lat pad mop telnet rlogin udptn nasi ssh
!
end
```

CHAPTER 26

Configuring PKI Using the IPSec VPN SPA

This chapter provides information about configuring PKI-related features using the IPSec VPN SPA on the Cisco 7600 series router. It includes the following sections:

Note The procedures in this chapter assume you have some familiarity with PKI configuration concepts. For detailed information about PKI configuration concepts, refer to the *Cisco IOS Security Configuration Guide* and the *Cisco IOS Security Command Reference*.

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Tip To ensure a successful configuration of your VPN using the IPSec VPN SPA, read all of the configuration summaries and guidelines before you perform any configuration tasks.

Overview of PKI

Cisco IOS public key infrastructure (PKI) provides certificate management to support security protocols such as IP Security (IPSec), secure shell (SSH), and secure socket layer (SSL).

A PKI is composed of the following entities:

- Peers communicating on a secure network
- At least one certificate authority (CA) that grants and maintains certificates
- Digital certificates, which contain information such as the certificate validity period, peer identity information, encryption keys that are used for secure communications, and the signature of the issuing CA
- An optional registration authority (RA) to offload the CA by processing enrollment requests
- A distribution mechanism (such as Lightweight Directory Access Protocol (LDAP) or HTTP) for certificate revocation lists (CRLs)

PKI provides customers with a scalable, secure mechanism for distributing, managing, and revoking encryption and identity information in a secured data network. Every entity (a person or a device) participating in the secured communications is enrolled in the PKI , a process where the entity generates a Rivest, Shamir, and Adelman (RSA) key pair (one private key and one public key) and has their identity validated by a trusted entity (also known as a CA or trustpoint).

After each entity enrolls in a PKI, every peer (also known as an end host) in a PKI is granted a digital certificate that has been issued by a CA. When peers must negotiate a secured communication session, they exchange digital certificates. Based on the information in the certificate, a peer can validate the identity of another peer and establish an encrypted session with the public keys contained in the certificate.

Configuring PKI involves the following tasks:

- Deploying Rivest, Shamir, and Adelman (RSA) keys within a public key infrastructure (PKI). An RSA key pair (a public and a private key) is required before you can obtain a certificate for your router; that is, the end host must generate a pair of RSA keys and exchange the public key with the certificate authority (CA) to obtain a certificate and enroll in a PKI.
- Configuring authorization and revocation of certificates within a PKI. After a certificate is validated as a properly signed certificate, it is authorized—via methods such as certificate maps, PKI-AAA, or a certificate-based access control list (ACL)—and the revocation status is checked by the issuing certificate authority (CA) to ensure that the certificate has not been revoked.

- Configuring certificate enrollment, which is the process of obtaining a certificate from a certificate authority (CA). Certificate enrollment occurs between the end host requesting the certificate and the CA. Each peer that participates in the public key infrastructure (PKI) must enroll with a CA. Various methods are available for certificate enrollment.
- Storing public key infrastructure (PKI) credentials, such as Rivest, Shamir, and Adelman (RSA) keys and certificates. These credentials can be stored in the default location on the router, which is NVRAM, or other locations.

Note As of Cisco IOS Release 12.2(33)SRA, a maximum number of 2000 IPSec tunnels is supported when PKI is configured with the IPSec VPN SPA.

Configuring RSA Key Pairs

A Cisco 7600 series router can have multiple Rivest, Shamir, and Adelman (RSA) key pairs. Thus, the Cisco IOS software can maintain a different key pair for each identity certificate.

RSA Key Pairs Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring RSA key pairs:

- It is recommended Secure Socket Layer (SSL) or other PKI clients do not attempt to enroll with the same CA multiple times.
- Internet Key Exchange (IKE) will not work for any identity that is configured to use a named key pair. If an IKE peer requests a certificate from a PKI trustpoint that is using multiple key support, the initial portion of the exchange will work, that is, the correct certificate will be sent in the certificate response; however, the named key pair will not be used and the IKE negotiation will fail.
- Whenever you regenerate a key pair, you must always re-enroll the certificate identities with that key pair.

To configure an RSA key pair, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto key generate rsa** [**usage-keys** | **general-keys**] [*key-pair-label*]	Generates RSA key pairs. • **usage-keys**—(Optional) Specifies that two special-usage key pairs should be generated, instead of one general-purpose key pair. • **general-keys**—(Optional) Specifies that the general-purpose key pair should be generated. • *key-pair-label*—(Optional) Specifies the name of the key pair that the router will use. (If this argument is enabled, you must specify either **usage-keys** or **general-keys**.)

	Command	Purpose
Step 2	Router(config)# **crypto pki trustpoint** *name*	Declares the CA that the router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 3	Router(ca-trustpoint)# **rsakeypair** *key-label* [*key-size* [*encryption-key-size*]]	Specifies which key pair to associate with the certificate. • *key-label*—The name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is configured. • *key-size*—(Optional) The size of the desired RSA key. If not specified, the existing key size is used. (The specified size must be the same as the *encryption-key-size*.) • *encryption-key-size*—(Optional) The size of the second key, which is used to request separate encryption, signature keys, and certificates. (The specified size must be the same as the *key-size*.)

Removing RSA Key Pair Settings

To delete a specified RSA key pair or all RSA key pairs that have been generated by your router, enter the **crypto key zeroize rsa** command in global configuration mode as follows:

Router(config)# **crypto key zeroize rsa** [*key-pair-label*]

In this command, *key-pair-label* specifies the name of the key pair to be deleted. If the *key-pair-label* argument is used, you delete only the specified RSA key pair. If no argument is used, you delete all the RSA key pairs from your router.

Verifying RSA Key Information

To verify RSA key information, use at least one of the following privileged EXEC commands.

To display your router's RSA public keys, use the **show crypto key mypubkey rsa** command:

```
Router# show crypto key mypubkey rsa

% Key pair was generated at: 06:07:50 UTC Jan 13 1996

Key name: myrouter.example.com

 Usage: Encryption Key

 Key Data:

  00302017 4A7D385B 1234EF29 335FC973 2DD50A37 C4F4B0FD 9DADE748 429618D5

  18242BA3 2EDFBDD3 4296142A DDF7D3D8 08407685 2F2190A0 0B43F1BD 9A8A26DB

  07953829 791FCDE9 A98420F0 6A82045B 90288A26 DBC64468 7789F76E EE21
```

To display a list of all the RSA public keys stored on your router (including the public keys of peers that have sent your router their certificates during peer authentication for IPSec), or to display details of a particular RSA public key stored on your router, use the **show crypto key pubkey-chain rsa** command:

```
Router# show crypto key pubkey-chain rsa

Codes: M - Manually Configured, C - Extracted from certificate

Code  Usage         IP-address     Name

M     Signature     10.0.0.1       myrouter.example.com

M     Encryption    10.0.0.1       myrouter.example.com

C     Signature     172.16.0.1     routerA.example.com

C     Encryption    172.16.0.1     routerA.example.com

C     General       192.168.10.3   routerB.domain1.com
```

For complete configuration information for Multiple RSA Key Pair Support, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/ftmltkey.htm

For an RSA key pair configuration example, see the "RSA Key Pairs Configuration Example" section on page 26-52.

Configuring Protected Private Key Storage

The Protected Private Key Storage feature allows a user to encrypt and lock the RSA private keys that are used on a Cisco 7600 series router, thereby preventing unauthorized use of the private keys.

Protected Private Key Storage Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Protected Private Key Storage:

- An encrypted key is not effective after the router boots up until you manually unlock the key (using the **crypto key unlock rsa** command). Depending on which key pairs are encrypted, this functionality may adversely affect applications such as IP Security (IPSec), Secure Shell (SSH) and Secure Socket Layer (SSL); that is, management of the router over a secure channel may not be possible until the necessary key pair is unlocked.
- If a passphrase is lost, you must regenerate the key, enroll with the CA server again, and obtain a new certificate. A lost passphrase cannot be recovered.
- If you want to change a passphrase, you must decrypt the key with the current passphrase using the **crypto key decrypt rsa** command and encrypt the key once more to specify the new passphrase.

Configuring Private Keys

To encrypt, decrypt, lock, and unlock private keys, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto key encrypt** [**write**] **rsa** [**name** *key-name*] **passphrase** *passphrase*	Encrypts the RSA keys. After this command is issued, the router can continue to use the key; the key remains unlocked. • **write**—(Optional) Router configuration is immediately written to NVRAM. If the write keyword is not issued, the configuration must be manually written to NVRAM; otherwise, the encrypted key will be lost next time the router is reloaded. • **name** *key-name*—(Optional) Name of the RSA key pair that is to be encrypted. If a key name is not specified, the default key name, routername.domainname, is used. • **passphrase** *passphrase*—Passphrase that is used to encrypt the RSA key. To access the RSA key pair, the passphrase must be specified.
Step 2	Router(config)# **exit**	Exits global configuration mode.
Step 3	Router# **show crypto key mypubkey rsa**	(Optional) Shows that the private key is encrypted (protected) and unlocked.
Step 4	Router# **crypto key lock rsa** [**name** *key-name*] **passphrase** *passphrase*	(Optional) Locks the encrypted private key on a running router. • **name** *key-name*—(Optional) Name of the RSA key pair that is to be locked. If a key name is not specified, the default key name, routername.domainname, is used. • **passphrase** *passphrase*—Passphrase that is used to lock the RSA key. To access the RSA key pair, the passphrase must be specified. **Note** After the key is locked, it cannot be used to authenticate the router to a peer device. This behavior disables any IPSec or SSL connections that use the locked key. Any existing IPSec tunnels created on the basis of the locked key will be closed. If all RSA keys are locked, SSH will automatically be disabled.
Step 5	Router# **show crypto key mypubkey rsa**	(Optional) Shows that the private key is protected and locked. The output will also show failed connection attempts by applications such as IKE, SSH, and SSL.

	Command	Purpose
Step 6	Router# **crypto key unlock rsa** [**name** *key-name*] **passphrase** *passphrase*	(Optional) Unlocks the private key. • **name** *key-name*—(Optional) Name of the RSA key pair that is to be unlocked. If a key name is not specified, the default key name, routername.domainname, is used. • **passphrase** *passphrase*—Passphrase that is used to unlock the RSA key. To access the RSA key pair, the passphrase must be specified. **Note** After this command is issued, you can continue to establish IKE tunnels.
Step 7	Router# **configure terminal**	Enters global configuration mode.
Step 8	Router(config)# **crypto key decrypt** [**write**] **rsa** [**name** *key-name*] **passphrase** *passphrase*	(Optional) Deletes the encrypted key and leaves only the unencrypted key. • **write**—(Optional) Unencrypted key is immediately written to NVRAM. If the write keyword is not issued, the configuration must be manually written to NVRAM; otherwise, the key will remain encrypted the next time the router is reloaded. • **name** *key-name*—(Optional) Name of the RSA key pair that is to be deleted. If a key name is not specified, the default key name, routername.domainname, is used. • **passphrase** *passphrase*—Passphrase that is used to delete the RSA key. To access the RSA key pair, the passphrase must be specified.

Verifying the Protected and Locked Private Keys

To verify that the key is protected (encrypted) and locked, enter the **show crypto key mypubkey rsa** command:

```
Router# show crypto key mypubkey rsa

% Key pair was generated at:20:29:41 GMT Jun 20 2003
Key name:pki1-72a.cisco.com
Usage:General Purpose Key
*** The key is protected and LOCKED. ***
Key is exportable.
Key Data:
305C300D 06092A86 4886F70D 01010105 00034B00 30480241 00D7808D C5FF14AC
0D2B55AC 5D199F2F 7CB4B355 C555E07B 6D0DECBE 4519B1F0 75B12D6F 902D6E9F
B6FDAD8D 654EF851 5701D5D7 EDA047ED 9A2A619D 5639DF18 EB020301 0001
```

For complete configuration information for Protected Private Key Storage, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/gt_ppkey.htm

For protected private key configuration examples, see the "Protected Private Key Storage Configuration Examples" section on page 26-53.

Configuring a Trustpoint CA

The **crypto pki trustpoint** command allows you to declare the certificate authority (CA) that your router should use and to specify characteristics for the CA.

Trustpoint CA Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring a trustpoint CA:

- After the trustpoint CA has been configured, you can obtain the certificate of the CA by using the **crypto pki authenticate** command or you can specify that certificates should not be stored locally but retrieved from a CA trustpoint by using the **crypto pki certificate query** command.
- Normally, certain certificates are stored locally in the router's NVRAM, and each certificate uses a moderate amount of memory. To save NVRAM space, you can use the **crypto pki certificate query** command to put the router into query mode, preventing certificates from being stored locally; instead, they are retrieved from a specified CA trustpoint when needed. This will save NVRAM space but could result in a slight performance impact.

To declare the CA that your router should use and specify characteristics for the trustpoint CA, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the CA that your router should use. Enabling this command puts you in ca-trustpoint configuration mode. • *name*—Name for the trustpoint CA.
Step 2	Router(ca-trustpoint)# **enrollment** [[**mode ra**] | [**retry period** *minutes*] | [**retry count** *number*] | [**url** *url*]]	Specifies enrollment parameters for your CA. • **mode ra**—(Optional) Specifies registration authority (RA) mode if your CA system provides a RA. RA mode is turned off until you enable the **mode ra** keyword. • **retry period** *minutes*—(Optional) Specifies the wait period between certificate request retries. The default is 1 minute between retries. (Specify from 1 to 60 minutes.) • **retry count** *number*—(Optional) Specifies the number of times a router will resend a certificate request when it does not receive a response from the previous request. The default is 10 retries. (Specify from 1 to 100 retries.) • **url** *url*—(Optional) Specifies the URL of the CA where your router should send certificate requests; for example, http://ca_server. *url* must be in the form http://CA_*name*, where CA_*name* is the CA's host Domain Name System (DNS) name or IP address.
	or	
	Router(ca-trustpoint)# **root tftp** *server-hostname filename*	Obtains the CA via TFTP. • *server-hostname*—Name for the server that will store the trustpoint CA. • *filename*—Name for the file that will store the trustpoint CA.
Step 3	Router(ca-trustpoint)# **enrollment http-proxy** *host-name port-num*	Obtains the CA via HTTP through the proxy server. • *host-name*—Name of the proxy server used to get the CA. • *port-num*—Port number used to access the CA. **Note** This command can be used in conjunction only with the **enrollment** command.
Step 4	Router(ca-trustpoint)# **primary** *name*	(Optional) Assigns a specified trustpoint as the primary trustpoint of the router. • *name*—Name of the primary trustpoint of the router.

	Command	Purpose
Step 5	Router(ca-trustpoint)# **crl** {**query** *url* | **optional**}	(Optional) Queries the certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked. • **query** *url*—Lightweight Directory Access Protocol (LDAP) URL published by the certificate authority (CA) server is specified to query the CRL; for example, ldap://another_server. • **optional**—CRL verification is optional. **Note** If the **query** *url* option is not enabled, the router will check the certificate distribution point (CDP) that is embedded in the certificate.
Step 6	Router(ca-trustpoint)# **default** *command-name*	(Optional) Sets the value of ca-trustpoint configuration mode to its default. • *command-name*—PKI-trustpoint configuration subcommand. Default is off.
Step 7	Router(ca-trustpoint)# **exit**	Exits ca-trustpoint configuration mode and enters global configuration mode.
Step 8	Router(config)# **crypto pki authenticate** *name*	Authenticates the CA (by obtaining the certificate of the CA.) • *name*—Name of the CA. Enter the *name* value entered in Step 1.
Step 9	Router(config)# **crypto pki trustpoint** *name*	Re-enters ca-trustpoint configuration mode. • *name*—Name for the trustpoint CA.
Step 10	Router(ca-trustpoint)# **crypto pki certificate query**	(Optional) Turns on query mode per specified trustpoint, causing certificates not to be stored locally.

Verifying a Trustpoint CA

To verify information about your certificate, the certificate of the CA, and registration authority (RA) certificates, enter the **show crypto pki certificates** command:

```
Router# show crypto pki certificates

CA Certificate

  Status: Available

  Certificate Serial Number: 3051DF7123BEE31B8341DFE4B3A338E5F

  Key Usage: Not Set

RA Signature Certificate

  Status: Available

  Certificate Serial Number: 34BCF8A0
```

```
  Key Usage: Signature

RA KeyEncipher Certificate

  Status: Available

  Certificate Serial Number: 34BCF89F

  Key Usage: Encryption
```

To display the trustpoints that are configured in the router, enter the **show crypto pki trustpoints** command:

```
Router# show crypto pki trustpoints

Trustpoint bo:
Subject Name:
CN = bomborra Certificate Manager
O = cisco.com
C = US
Serial Number:01
Certificate configured.
CEP URL:http://bomborra
CRL query url:ldap://bomborra
```

For complete configuration information for the trustpoint CA, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/fttrust.htm

For a trustpoint CA configuration example, see the "Trustpoint CA Configuration Example" section on page 26-53.

Configuring Query Mode Definition Per Trustpoint

Certificates contain public key information and are signed by certificate authority (CA) as proof of identity. Normally, all certificates are stored locally in the router's NVRAM, and each certificate uses a moderate amount of memory. The Query Mode Definition Per Trustpoint feature allows you to define a query for a specific trustpoint so that the certificates associated with that specific trustpoint can be stored on a remote server.

This feature is especially useful for environments where multiple trustpoints are configured on a router because it allows you more control over use of the trustpoint. Query mode can be activated on specific trustpoints rather than on all of the trustpoints on a router.

Query Mode Definition Per Trustpoint Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Query Mode Definition Per Trustpoint:

- Normally, certain certificates are stored locally in the router's NVRAM, and each certificate uses a moderate amount of memory. To save NVRAM space, you can use the **query certificate** command to prevent certificates from being stored locally; instead, they are retrieved from a remote server, such as a CA or LDAP server, during startup. This will save NVRAM space but could result in a slight performance impact.

- Certificates associated with a specified trustpoint will not be written into NVRAM and the certificate query will be attempted during the next reload of the router.
- When the global **crypto pki certificate query** command is used, the query certificate will be added to all trustpoints on the router. When the **no crypto pki certificate query** command is used, any previous query certificate configuration will be removed from all trustpoints and any query in progress will be halted and the feature disabled.

To configure a trustpoint CA and initiate query mode for the trustpoint, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoin**t *name*	Declares the CA that your router should use. Enabling this command puts you in ca-trustpoint configuration mode. • *name*—Name for the trustpoint CA.
Step 2	Router(ca-trustpoint)# **enrollment** [[**mode ra**] | [**retry period** *minutes*] | [**retry count** *number*] | [**url** *url*]]	Specifies enrollment parameters for your CA. • **mode ra**—(Optional) Specifies registration authority (RA) mode if your CA system provides a RA. RA mode is turned off until you enable the **mode ra** keyword. • **retry period** *minutes*—(Optional) Specifies the wait period between certificate request retries. The default is 1 minute between retries. (Specify from 1 to 60 minutes.) • **retry** *count number*—(Optional) Specifies the number of times a router will resend a certificate request when it does not receive a response from the previous request. The default is 10 retries. (Specify from 1 to 100 retries.) • **url** *url*—(Optional) Specifies the URL of the CA where your router should send certificate requests; for example, http://ca_server. *url* must be in the form http://CA_*name*, where CA_*name* is the CA's host Domain Name System (DNS) name or IP address.
Step 3	Router(ca-trustpoint)# **enrollment http-proxy** *host-name port-num*	(Optional) Obtains the CA via HTTP through the proxy server. • *host-name*—Name of the proxy server used to get the CA. • *port-num*—Port number used to access the CA. **Note** This command can be used in conjunction only with the **enrollment** command.
Step 4	Router(ca-trustpoint)# **crl query** *url*	(Optional) Specifies the URL for the CA server if the CA server supports query mode through LDAP. • *url*—Lightweight Directory Access Protocol (LDAP) URL published by the certificate authority (CA) server.

	Command	Purpose
Step 5	Router(ca-trustpoint)# **default** *command-name*	(Optional) Sets the value of ca-trustpoint configuration mode to its default. • *command-name*—PKI-trustpoint configuration subcommand. Default is off.
Step 6	Router(ca-trustpoint)# **query certificate**	Turns on query mode per specified trustpoint, causing certificates not to be stored locally and to be retrieved from a remote server.
Step 7	Router(ca-trustpoint)# **exit**	Exits ca-trustpoint configuration mode and enters global configuration mode.
Step 8	Router(config)# **crypto pki authenticate** *name*	Authenticates the CA (by obtaining the certificate of the CA.) • *name*—Name of the CA. Enter the *name* value entered in Step 1.
Step 9	Router(config)# **crypto key generate rsa**	(Optional) Generates RSA key pairs.
Step 10	Router(config)# **crypto pki enroll** *trustpoint-name*	(Optional) Obtains router certificate. • *trustpoint-name*—Name of the CA. Enter the *name* value entered in Step 1.

Verifying Query Mode Definition Per Trustpoint CA

For query mode to operate correctly during the next reload, the certificates must be associated with the trustpoint. Use the **show crypto pki certificates** command to verify that each of the trustpoints has the needed certificates before storing the configuration and reloading the router:

```
Router# show crypto pki certificates status

Trustpoint yni:

  Issuing CA certificate pending:

    Subject Name:

     cn=nsca-r1 Cert Manager,ou=pki,o=cisco.com,c=US

    Fingerprint: C21514AC 12815946 09F635ED FBB6CF31

  Router certificate pending:

    Subject Name:

     hostname=trance.cisco.com,o=cisco.com

  Next query attempt:

    52 seconds
```

For complete configuration information for Query Mode Definition Per Trustpoint, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/gt_qerym.htm

For a query mode definition per trustpoint configuration example, see the "Query Mode Definition Per Trustpoint Configuration Example" section on page 26-53.

Configuring a Local Certificate Storage Location

The Local Certificate Storage Location feature enables you to store public key infrastructure (PKI) credentials, such as Rivest, Shamir, and Adelman (RSA) keys and certificates in a specific location. An example of a certificate storage location includes NVRAM, which is the default location, and other local storage locations as supported by your platform, such as flash.

Note The Local Certificate Storage Location feature is only supported as of Cisco IOS Release 12.2(33)SRA.

Local Certificate Storage Location Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring a local certificate storage location:

- Before you can specify the local certificate storage location, your system should meet the following requirements:
  - A Cisco IOS Release 12.4(2)T PKI-enabled image or a later image
  - A platform that supports storing PKI credentials as separate files
  - A configuration that contains at least one certificate
  - An accessible local file system
- When storing certificates to a local storage location, the following restrictions are applicable:
  - Only local file systems may be used. An error message will be displayed if a remote file system is selected, and the command will not take effect.
  - A subdirectory may be specified if supported by the local file system. NVRAM does not support subdirectories.
  - Certificates are stored to NVRAM by default, however, some routers do not have the required amount of NVRAM to successfully store certificates. Introduced in Cisco IOS Release 12.4(2)T is the ability to specify where certificates are stored on a local file system.
  - During run time, you can specify what active local storage device you would like to use to store certificates.

Specifying a Local Storage Location for Certificates

To specify the local storage location for certificates, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki certificate storage** *location-name*	Specifies the local storage location for certificates. • *location-name*—Name of the storage location.

	Command	Purpose
Step 2	Router (config)# **exit**	Exits global configuration mode.
Step 3	Router# **copy** *source-url destination-url*	(Optional) Saves the running configuration to the startup configuration. • *source-url*—The location URL (or alias) of the source file or directory to be copied. The source can be either local or remote, depending upon whether the file is being downloaded or uploaded. • *destination-url*—The destination URL (or alias) of the copied file or directory. The destination can be either local or remote, depending upon whether the file is being downloaded or uploaded. **Note** Settings will only take effect when the running configuration is saved to the startup configuration.

Verifying the Local Certificate Storage Location Configuration

To verify a local certificate storage location configuration, enter the **show crypto pki certificates storage** command.

The **show crypto pki certificates storage** command displays the current setting for the PKI certificate storage location.

The following example shows that certificates are stored in the certs subdirectory of disk0:

```
Router# show crypto pki certificates storage

Certificates will be stored in disk0:/certs/
```

For complete configuration information for local certificate storage location, refer to the *Cisco IOS Security Configuration Guide* or the following URL:

http://www.cisco.com/en/US/products/ps6441/products_feature_guide09186a00804a5a7f.html

For local certificate storage configuration examples, see the "Local Certificate Storage Location Configuration Example" section on page 26-54.

Configuring Direct HTTP Enroll with CA Servers (Re-enroll Using Existing Certificates)

The Direct HTTP Enroll with CA Servers feature allows users to bypass the registration authority (RA) when enrolling with a certificate authority (CA) by configuring an enrollment profile. Thus, HTTP enrollment requests can be sent directly to the CA server.

The "re-enroll using existing certificates" functionality allows a router that is enrolled with a third-party vendor CA to use its existing certificate to enroll with the Cisco IOS certificate server so the enrollment request is automatically granted.

Direct HTTP Enroll with CA Servers Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Direct HTTP Enroll with CA Servers:

- The CA certificate and router certificates must be returned in the privacy enhanced mail (PEM) format.
- If an enrollment profile is specified, an enrollment URL cannot be specified in the trustpoint configuration.
- Because there is no standard for the HTTP commands used by various CAs, the user is required to enter the command that is appropriate to the CA that is being used.
- The newly created trustpoint can only be used one time (which occurs when the router is enrolled with the Cisco IOS CA). After the initial enrollment is successfully completed, the credential information will be deleted from the enrollment profile.
- The Cisco IOS certificate server will automatically grant only the requests from clients who were already enrolled with the non-Cisco IOS CA. All other requests must be manually granted—unless the server is set to be in autogrant mode (using the **grant automatic** command).
- To configure Direct HTTP Enroll with CA Servers, you must perform the following steps:
  - Either configure a certificate enrollment profile for the client router (see the "Configuring an Enrollment Profile for a Client Router" section on page 26-16) or configure an enrollment profile for a client router that is already enrolled with a third-party vendor (see the "Configuring an Enrollment Profile for a Client Router Enrolled with a Third-Party Vendor CA" section on page 26-18).
  - Configure the CA certificate server to accept enrollment requests only from clients who are already enrolled with the third-party vendor CA trustpoint (see the "Configuring the CA to Accept Enrollment Requests from Clients of a Third-Party Vendor CA" section on page 26-19).

Configuring an Enrollment Profile for a Client Router

To configure a certificate enrollment profile, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the trustpoint a given name and enters ca-trustpoint configuration mode. • *name*—Name of the CA trustpoint.
Step 2	Router(ca-trustpoint)# **enrollment profile** *label*	Specifies that an enrollment profile can be used for certificate authentication and enrollment. • *label*—Name for the enrollment profile.
Step 3	Router(ca-trustpoint)# **exit**	Exits ca-trustpoint configuration mode and enters global configuration mode.
Step 4	Router(config)# **crypto pki profile enrollment** *label*	Defines an enrollment profile and enters ca-profile-enroll configuration mode. • *label*—Name for the enrollment profile; the enrollment profile name must match the name specified in the **enrollment profile** command.

	Command	Purpose
Step 5	Router(ca-profile-enroll)# **authentication url** *url* or Router(ca-profile-enroll)# **authentication terminal**	(Optional) Specifies the URL of the CA server to which to send certificate authentication requests. • *url*—URL of the CA server to which your router should send authentication requests. If using HTTP, the URL should read "http://CA_name," where CA_name is the host Domain Name System (DNS) name or IP address of the CA. If using TFTP, the URL should read "tftp://certserver/file_specification." If the URL does not include a file specification, the fully qualified domain name (FQDN) of the router will be used. Specifies manual cut-and-paste certificate authentication.
Step 6	Router(ca-profile-enroll)# **authentication command**	(Optional) Sends the HTTP request to the CA for authentication. This command should be used after the **authentication url** command has been entered.
Step 7	Router(ca-profile-enroll)# **enrollment url** *url* or Router(ca-profile-enroll)# **enrollment terminal**	Specifies the URL of the CA server to which to send certificate enrollment requests via HTTP or TFTP. • *url*—URL of the CA server. Specifies manual cut-and-paste certificate enrollment.
Step 8	Router(ca-profile-enroll)# **enrollment command**	(Optional) Specifies the HTTP command is sent to the CA for enrollment.
Step 9	Router(ca-profile-enroll)# **parameter** *number* {**value** *value* | **prompt** *string*}	(Optional) Specifies parameters for an enrollment profile. • *number*—User parameters. Valid values range from 1 to 18. • *value*—To be used if the parameter has a constant value. • *string*—To be used if the parameter is supplied after the **crypto pki authenticate** command or the **crypto pki enroll** command has been entered. **Note** The value of the *string* argument does not have an effect on the value that is used by the router. This command can be used multiple times to specify multiple values.
Step 10	Router(ca-profile-enroll config)# **exit**	Exits ca-profile-enroll configuration mode and enters global configuration mode.

	Command	Purpose
Step 11	Router(config)# **exit**	Exits global configuration mode and enters privileged EXEC mode.
Step 12	Router# **show crypto pki certificates**	(Optional) Verifies information about your certificate, the certificate of the CA, and RA certificates.
Step 13	Router# **show crypto pki trustpoints**	(Optional) Displays the trustpoints that are configured in the router.

Configuring an Enrollment Profile for a Client Router Enrolled with a Third-Party Vendor CA

When a client router is already enrolled with a third-party vendor CA, but you want to re-enroll that router with a Cisco IOS certificate server, perform the following procedures. Note that some prerequisite steps are required before beginning the configuration.

Prerequisites

Before configuring a certificate enrollment profile for the client router enrolled with a third-party vendor, you should have already performed the following tasks at the client router:

- Defined a trustpoint that points to a third-party vendor CA.
- Authenticated and enrolled the client router with the third-party vendor CA.

To configure a certificate enrollment profile for a client router that is already enrolled with a third-party vendor CA so that the router can re-enroll with a Cisco IOS certificate server, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the Cisco IOS CA that is to be used.
Step 2	Router(ca-trustpoint)# **enrollment profile** *label*	Specifies that an enrollment profile is to be used for certificate reenrollment. • *label*—Name for the enrollment profile.
Step 3	Router(ca-trustpoint)# **exit**	Exists ca-trustpoint configuration mode and enters global configuration mode.
Step 4	Router(config)# **crypto pki profile enrollment** *label*	Defines an enrollment profile and enters ca-profile-enroll configuration mode. • *label*—Name for the enrollment profile; the enrollment profile name must match the name specified in the **enrollment profile** command in Step 2.

	Command	Purpose
Step 5	Router(ca-profile-enroll)# **enrollment url** *url*	Specifies the URL of the CA server to which to send certificate enrollment requests via HTTP. • *url*—The enrollment URL should point to the Cisco IOS CA.
Step 6	Router(ca-profile-enroll)# **enrollment credential** *label*	Specifies the non-Cisco IOS CA trustpoint that is to be enrolled with the Cisco IOS CA. • *label*—Name of the CA trustpoint of another vendor.
Step 7	Router(ca-profile-enroll)# **exit**	Exits ca-profile-enroll configuration mode and enters global configuration mode.
Step 8	Router(config)# **exit**	Exits global configuration mode and enters privileged EXEC mode.
Step 9	Router# **show crypto pki certificates**	(Optional) Verifies information about your certificate, the certificate of the CA, and RA certificates
Step 10	Router# **show crypto pki trustpoints**	(Optional) Displays the trustpoints that are configured in the router.

Configuring the CA to Accept Enrollment Requests from Clients of a Third-Party Vendor CA

To configure the CA certificate server to accept enrollment requests only from clients who are already enrolled with the third-party vendor CA trustpoint, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **ip http server**	Enables the HTTP server on your system.
Step 2	Router(config)# **crypto pki server** *cs-label*	Enables the certificate server and enters certificate server configuration mode. • *cs-label*—The *cs-label* argument must match the name that was specified by the **crypto pki trustpoint** command for the client router.
Step 3	Router(cs-server)# **database url** *root-url*	Specifies the location where all database entries for the certificate server will be written out. • *root-url*—Root URL. **Note** If this command is not specified, all database entries will be written to NVRAM.

	Command	Purpose
Step 4	Router(cs-server)# **database level** {**minimal** | **names** | **complete**}	Controls what type of data is stored in the certificate enrollment database. • **minimal**—Enough information is stored only to continue issuing new certificates without conflict; the default value. • **names**—In addition to the information given in the minimal level, the serial number and subject name of each certificate. • **complete**—In addition to the information given in the minimal and names levels, each issued certificate is written to the database. **Note** The **complete** keyword produces a large amount of information; if it is issued, you should also specify an external TFTP server in which to store the data using the **database url** command.
Step 5	Router(cs-server)# **issuer-name** *DN-string*	Sets the CA issuer name to the specified DN-string. • *DN-string*—The default value is as follows: **issuer-name CN=cs**-*label*.
Step 6	Router(cs-server)# **grant auto trustpoint** *label*	Enables the certificate server to automatically grant only the requests from clients that are already enrolled with the specified non-Cisco IOS CA trustpoint. • *label*—Name of the CA trustpoint of another vendor. **Note** The *label* argument should match the trustpoint that was specified for the client router's enrollment profile (using the **enrollment credential** command).
Step 7	Router(cs-server)# **lifetime** {**ca-certificate** | **certificate**} *time*	(Optional) Specifies the lifetime, in days, of a CA certificate or a certificate. • *time*—Valid values range from 1 day to 1825 days. The default CA certificate lifetime is 3 years; the default certificate lifetime is 1 year. The maximum certificate lifetime is 1 month less than the lifetime of the CA certificate.
Step 8	Router(cs-server)# **lifetime crl** *time*	(Optional) Defines the lifetime, in hours, of the Certificate Revocation List (CRL) that is used by the certificate server. • *time*—Maximum lifetime value is 336 hours (2 weeks). The default value is 168 hours (1 week).
Step 9	Router(cs-server)# **cdp-url** *url*	(Optional) Defines a Certificate Distribution Point (CDP) to be used in the certificates that are issued by the certificate server. • *url*—URL must be an HTTP URL.

	Command	Purpose
Step 10	Router(cs-server)# **shutdown**	Disables a certificate server without removing the configuration. You should issue this command only after you have completely configured your certificate server.
Step 11	Router(cs-server)# **exit**	Exits certificate server configuration mode.
Step 12	Router(config)# **exit**	Exits global configuration mode.
Step 13	Router# **show crypto pki server**	(Optional) Displays the current state and configuration of the certificate server.

For complete configuration information for Direct HTTP Enroll with CA Servers, including the "re-enroll using existing certificates" functionality, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122limit/122z/122zh13/gthttpca.htm

For Direct HTTP Enroll with CA Servers configuration examples, see the "Direct HTTP Enrollment with CA Servers Configuration Examples" section on page 26-54.

Configuring Manual Certificate Enrollment (TFTP and Cut-and-Paste)

The Manual Certificate Enrollment (TFTP and Cut-and-Paste) feature allows users to generate a certificate request and accept certificate authority (CA) certificates as well as the router's certificates; these tasks are accomplished by a TFTP server or manual cut-and-paste operations. Users may wish to utilize TFTP or manual cut-and-paste enrollment in the following situations:

- The CA does not support Simple Certificate Enrollment Protocol (SCEP) (which is the most commonly used method for sending and receiving requests and certificates)
- A network connection between the router and CA is not possible (which is how a router running Cisco IOS software obtains its certificate)

Manual Certificate Enrollment (TFTP and Cut-and-Paste) Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Manual Certificate Enrollment (TFTP and Cut-and-Paste):

- A user can switch between TFTP and cut-and-paste; for example, a user can paste the CA certificate using the **enrollment terminal** command, then enter **no enrollment terminal** and **enrollment url tftp://certserver/file_specification** to switch to TFTP to send or receive requests and router certificates. However, we do not recommend switching URLs if SCEP is used; that is, if the enrollment URL is "http://," do not change the enrollment URL between fetching the CA certificate and enrolling the certificate.

Configuring Manual Enrollment Using TFTP

Before configuring manual enrollment using TFTP, you must meet the following prerequisites:

- You must know the correct URL to use if you are configuring certificate enrollment using TFTP.
- The router must be able to write a file to the TFTP server for the **crypto pki enroll** command.
- Some TFTP servers require that the file exist on the server before it may be written.
- Most TFTP servers require that the file be "write-able" by the world. This requirement may pose a risk because any router or other device may write or overwrite the certificate request; thus, the router will not be able to use the certificate once it is granted by the CA because the request was modified.

To declare the trustpoint CA that your router should use and configure that trustpoint CA for manual enrollment using TFTP, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 2	Router(ca-trustpoint)# **enrollment** [**mode**] [**retry period** *minutes*] [**retry count** *number*] **url** *url*	Specifies the enrollment parameters of your CA. • **mode**—Specifies registration authority (RA) mode if your CA system provides a RA. • **retry period** *minutes*—Specifies the wait period between certificate request retries. The default is 1 minute between retries. • **retry count** *number*—Specifies the number of times a router will resend a certificate request when it does not receive a response from the previous request. (Specify from 1 to 100 retries.) • **url** *url*—Specifies the URL of the CA where your router should send certificate requests. If you are using SCEP for enrollment, the URL must be in the form http://CA_name, where CA_name is the CA's host Domain Name System (DNS) name or IP address. If you are using TFTP for enrollment, the URL must be in the form tftp://certserver/file_specification.
Step 3	Router(ca-trustpoint)# **crypto pki authenticate** *name*	Authenticates the CA (by obtaining the certificate of the CA.) • *name*—Name of the CA. Enter the *name* value entered in Step 1.
Step 4	Router(ca-trustpoint)# **exit**	Exits ca-trustpoint configuration mode and returns to global configuration.

	Command	Purpose
Step 5	Router(config)# **crypto pki enroll** *name*	Obtains your router's certificates from the CA. • *name*—Name of the CA. Enter the *name* value entered in Step 1.
Step 6	Router(config)# **crypto pki import** *name* **certificate**	Imports a certificate using TFTP. • *name*—Name of the CA. Enter the *name* value entered in Step 1.

Configuring Certificate Enrollment Using Cut-and-Paste

To declare the trustpoint CA that your router should use and configure that trustpoint CA for manual enrollment using cut-and-paste, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 2	Router(ca-trustpoint)# **enrollment terminal**	Specifies manual cut-and-paste certificate enrollment.
Step 3	Router(ca-trustpoint)# **crypto pki authenticate** *name*	Authenticates the CA (by obtaining the certificate of the CA). • *name*—Specifies the name of the CA. Enter the *name* value entered in Step 1.
Step 4	Router(ca-trustpoint)# **exit**	Exits ca-trustpoint configuration mode and returns to global configuration.
Step 5	Router(config)# **crypto pki enroll** *name*	Obtains your router's certificates from the CA. • *name*—Specifies the name of the CA. Enter the *name* value entered in Step 1.
Step 6	Router(config)# **crypto pki import** *name* **certificate**	Imports a certificate manually at the terminal. • *name*—Specifies the name of the CA. Enter the *name* value entered in Step 1. **Note** You must enter the **crypto pki import** command twice if usage keys (signature and encryption keys) are used. The first time the command is entered, one of the certificates is pasted into the router; the second time the command is entered, the other certificate is pasted into the router. (It does not matter which certificate is pasted first.)

Verifying the Manual Certificate Enrollment Configuration

To verify information about your certificate, the certificate of the CA, and RA certificates, enter the **show crypto pki certificates** command:

```
Router# show crypto pki certificates

Certificate
  Status:Available
  Certificate Serial Number:14DECE05000000000C48
  Certificate Usage:Encryption

  Issuer:
    CN = msca-root
    O = Cisco Systems
    C = U

  Subject:
    Name:Router.cisco.com
    OID.1.2.840.113549.1.9.2 = Router.cisco.com

     CRL Distribution Point:
    http://msca-root/CertEnroll/msca-root.crl

     Validity Date:
    start date:18:16:45 PDT Jun 7 2002
    end   date:18:26:45 PDT Jun 7 2003
    renew date:16:00:00 PST Dec 31 1969

     Associated Trustpoints:MS

   Certificate
    Status:Available
    Certificate Serial Number:14DEC2E9000000000C47
    Certificate Usage:Signature

     Issuer:
    CN = msca-root
    O = Cisco Systems
    C = US

     Subject:
    Name:Router.cisco.com
    OID.1.2.840.113549.1.9.2 = Router.cisco.com

     CRL Distribution Point:
        http://msca-root/CertEnroll/msca-root.crl

     Validity Date:
    start date:18:16:42 PDT Jun 7 2002
    end   date:18:26:42 PDT Jun 7 2003
    renew date:16:00:00 PST Dec 31 1969

     Associated Trustpoints:MS

   CA Certificate
    Status:Available
    Certificate Serial Number:3AC0A65E9547C2874AAF2468A942D5EE
    Certificate Usage:Signature

     Issuer:
    CN = msca-root
    O = Cisco Systems
```

```
    C = US

     Subject:
    CN = msca-root
    O = Cisco Systems
    C = US

     CRL Distribution Point:
        http://msca-root/CertEnroll/msca-root.crl

  Validity Date:
    start date:16:46:01 PST Feb 13 2002
    end   date:16:54:48 PST Feb 13 2007

  Associated Trustpoints:MS
```

To display the trustpoints that are configured in the router, enter the **show crypto pki trustpoints** command:

```
Router# show crypto pki trustpoints

Trustpoint bo:

    Subject Name:

    CN = bomborra Certificate Manager

     O = cisco.com

     C = US

          Serial Number:01

    Certificate configured.

    CEP URL:http://bomborra

    CRL query url:ldap://bomborra
```

For complete configuration information for Manual Certificate Enrollment (TFTP and Cut-and-Paste), refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t13/ftmancrt.htm

For manual certificate enrollment configuration examples, see the "Manual Certificate Enrollment Configuration Examples" section on page 26-56.

Configuring Certificate Autoenrollment

The Certificate Autoenrollment feature allows you to configure your router to automatically request a certificate from the certificate authority (CA) that is using the parameters in the configuration. Thus, operator convention is no longer required at the time the enrollment request is sent to the CA server.

Automatic enrollment will be performed on startup for any trustpoint CA that is configured and does not have a valid certificate. When the certificate—which is issued by a trustpoint CA that has been configured for autoenrollment—expires, a new certificate is requested. Although this feature does not provide seamless certificate renewal, it does provide unattended recovery from expiration.

Before the Certificate Autoenrollment feature, certificate enrollment required complicated, interactive commands that had to be executed on every router. This feature allows you to preload all of the necessary information into the configuration and cause each router to obtain certificates automatically when it is booted. Autoenrollment also checks for expired router certificates.

Note Before submitting an automatic enrollment request, all necessary enrollment information must be configured.

To configure autoenrollment with a CA on startup, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the name of the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 2	Router(ca-trustpoint)# **enrollment url** *url*	Specifies the URL of the CA on which your router should send certificate requests; for example, http://ca_server. • *url*—Must be in the form of http://CA_name, where CA_name is the name of the CA's host Domain Name System or the IP address.
Step 3	Router(ca-trustpoint)# **subject-name** [*x.500-name*]	(Optional) Specifies the requested subject name that will be used in the certificate request. • *x.500-name*—If the *x.500-name* argument is not specified, the fully qualified domain name (FQDN), which is the default subject name, is used.
Step 4	Router(ca-trustpoint)# **ip-address** {*interface* | **none**}	Includes the IP address of the specified interface in the certificate request. • *interface*—IP address of the interface. • **none**—Specify this keyword if no IP address should be included. If this command is enabled, you will not be prompted for an IP address during enrollment for this trustpoint.

	Command	Purpose
Step 5	Router(ca-trustpoint)# **serial-number** [**none**]	Specifies the router serial number in the certificate request, unless the **none** keyword is issued. • **none**—(Optional) Specify this keyword if no serial number should be included.
Step 6	Router(ca-trustpoint)# **auto-enroll** [**regenerate**]	Enables autoenrollment. This command allows you to automatically request a router certificate from the CA. By default, only the DNS name of the router is included in the certificate. • **regenerate**—(Optional) Specify this keyword to generate a new key for the certificate even if a named key already exists.
Step 7	Router(ca-trustpoint)# **password** *string*	(Optional) Specifies the revocation password for the certificate. • *string*—Text of the password. **Note** If this command is enabled, you will not be prompted for a password during enrollment for this trustpoint.
Step 8	Router(ca-trustpoint)# **rsakeypair** *key-label* [*key-size* [*encryption-key-size*]]	Specifies which key pair to associate with the certificate. • *key-label*—Name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is configured. • *key-size*—(Optional) Size of the desired RSA key. If not specified, the existing key size is used. (The specified size must be the same as the *encryption-key-size*.) • *encryption-key-size*—(Optional) Size of the second key, which is used to request separate encryption, signature keys, and certificates. (The specified size must be the same as the *key-size*.) If this command is not enabled, the FQDN key pair is used.

Preloading Root CAs

After enabling automatic enrollment, you must authenticate the CA to establish a chain of trust. This can be done by implementing one of the following methods:

- Getting the Certificate of the CA, page 26-28
- Adding the Certificate of the CA, page 26-28

Getting the Certificate of the CA

To get the certificate of the CA, enter the **crypto pki authenticate** command in global configuration mode.

```
Router(config)# crypto pki authenticate name
```

In this command, *name* specifies the name of the CA.

Adding the Certificate of the CA

To add the certificate of the CA, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router (config)# **crypto pki certificate chain** *name*	Enters certificate chain configuration mode, which allows you to add or delete specified certificates. • *name*—Name of the CA.
Step 2	Router (config-cert-chain)# **certificate** *certificate-serial-number*	Manually adds or deletes certificates. • *certificate-serial-number*—Serial number of the CA to add.

Verifying CA Information

To display information about your certificates, the certificates of the CA, and registration authority (RA) certificates, enter the **show crypto pki certificates** command:

```
Router# show crypto pki certificates

Certificate

  Subject Name

    Name: myrouter.example.com
    IP Address: 10.0.0.1

  Status: Available

  Certificate Serial Number: 428125BDA34196003F6C78316CD8FA95

  Key Usage: Signature


Certificate

  Subject Name

    Name: myrouter.example.com
    IP Address: 10.0.0.1

  Status: Available

  Certificate Serial Number: AB352356AFCD0395E333CCFD7CD33897

  Key Usage: Encryption
```

```
CA Certificate
  Status: Available
  Certificate Serial Number: 3051DF7123BEE31B8341DFE4B3A338E5F
  Key Usage: Not Set
```

To display the trustpoints configured in the router, enter the **show crypto pki trustpoints** command:

```
Router# show crypto pki trustpoints
Trustpoint bo:
    Subject Name:
    CN = bomborra Certificate Manager
     O = cisco.com
     C = US
    Serial Number:01
    Certificate configured.
    CEP URL:http://bomborra
    CRL query url:ldap://bomborra
```

For complete configuration information for Certificate Autoenrollment, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/ftautoen.htm

For a certificate autoenrollment configuration example, see the "Certificate Autoenrollment Configuration Example" section on page 26-59.

Configuring Key Rollover for Certificate Renewal

Automatic certificate enrollment was introduced to allow the router to automatically request a certificate from the certificate authority (CA) server. By default, the automatic enrollment feature requests a new certificate when the old certificate expires. Connectivity can be lost while the request is being serviced because the existing certificate and key pairs are deleted immediately after the new key is generated. The new key does not have a certificate to match it until the process is complete, and incoming Internet Key Exchange (IKE) connections cannot be established until the new certificate is issued. The Key Rollover for Certificate Renewal feature allows the certificate renewal request to be made before the certificate expires and retains the old key and certificate until the new certificate is available.

Key rollover can also be used with a manual certificate enrollment request. Using the same method as key rollover with certificate autoenrollment, a new key pair is created with a temporary name, and the old certificate and key pair are retained until a new certificate is received from the CA. When the new certificate is received, the old certificate and key pair are discarded and the new key pair is renamed with the name of the original key pair. Do not regenerate the keys manually; key rollover will occur when the **crypto pki enroll** command is issued.

Key Rollover for Certificate Renewal Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Key Rollover for Certificate Renewal:

- Trustpoints configured to generate a new key pair using the **regenerate** command or the **regenerate** keyword of the **auto-enroll** command must not share key pairs with other trustpoints. To give each trustpoint its own key pair, use the **rsakeypair** command in ca-trustpoint configuration mode. Sharing key pairs among regenerating trustpoints is not supported and will cause loss of service on some of the trustpoints because of key and certificate mismatch.

Configuring Automatic Certificate Enrollment with Key Rollover

To configure key rollover with automatic certificate enrollment, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the name of the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 2	Router(ca-trustpoint)# **enrollment url** *url*	Specifies the URL of the CA on which your router should send certificate requests; for example, http://ca_server. • *url*—Must be in the form of http://CA_name, where CA_name is the name of the CA's host Domain Name System or the IP address.
Step 3	Router(ca-trustpoint)# **subject-name** [*x.500-name*]	(Optional) Specifies the requested subject name that will be used in the certificate request. • *x.500-name*—(Optiona) If the *x.500-name* argument is not specified, the fully qualified domain name (FQDN), which is the default subject name, is used.
Step 4	Router(ca-trustpoint)# **ip-address** {*interface* | **none**}	Includes the IP address of the specified interface in the certificate request. • *interface*—IP address of the interface. • **none**—Specify this keyword if no IP address should be included. If this command is enabled, you will not be prompted for an IP address during enrollment for this trustpoint.
Step 5	Router(ca-trustpoint)# **serial-number** [**none**]	Specifies the router serial number in the certificate request, unless the **none** keyword is issued. • **none**—(Optional) Specify this keyword if no serial number should be included.

	Command	Purpose
Step 6	Router(ca-trustpoint)# **auto-enroll** [*percent*][**regenerate**]	Enables autoenrollment. This command allows you to automatically request a router certificate from the CA. By default, only the DNS name of the router is included in the certificate. • *percent*—(Optional) Use the *percent* argument to specify that a new certificate will be requested after the percent lifetime of the current certificate is reached. • **regenerate**—(Optional) Specify this keyword to generate a new key for the certificate even if a named key already exists. **Note** If the key pair being rolled over is exportable, the new key pair will also be exportable. The following comment will appear in the trustpoint configuration to indicate whether the key pair is exportable: `! RSA key pair associated with trustpoint is exportable.`
Step 7	Router(ca-trustpoint)# **password** *string*	(Optional) Specifies the revocation password for the certificate. • *string*—Text of the password. **Note** If this command is enabled, you will not be prompted for a password during enrollment for this trustpoint.
Step 8	Router(ca-trustpoint)# **rsakeypair** *key-label* [*key-siz*e [*encryption-key-size*]]	Specifies which key pair to associate with the certificate. • *key-label*—Name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is configured. • *key-size*—(Optional) Size of the desired RSA key. If not specified, the existing key size is used. (The specified size must be the same as the e*ncryption-key-size*.) • *encryption-key-size*—(Optional) Size of the second key, which is used to request separate encryption, signature keys, and certificates. (The specified size must be the same as the *key-size*.) **Note** If this command is not enabled, the FQDN key pair is used.
Step 9	Router(ca-trustpoint)# **exit**	Exits ca-trustpoint configuration mode and returns to global configuration mode.

	Command	Purpose
Step 10	Router(config)# **crypto pki authenticate** *name*	Authenticates the CA (by obtaining the certificate of the CA.) • *name*—Name of the CA. Enter the *name* value entered in Step 1. Check the certificate fingerprint if prompted. **Note** This command is optional if the CA certificate is already loaded into the configuration.
Step 11	Router(config)# **exit**	Exits global configuration mode and returns to privileged EXEC mode.
Step 12	Router# **copy system:running-config nvram:startup-config**	(Optional) Copies the running configuration to the NVRAM startup configuration.

Configuring Manual Certificate Enrollment with Key Rollover

To configure key rollover with manual certificate enrollment, perform the following steps beginning in global configuration mode:

Note Do not regenerate the keys manually using the **crypto key generate** command; key rollover will occur when the **crypto pki enroll** command is issued.

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the name of the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 2	Router(ca-trustpoint)# **enrollment url** *url*	Specifies the URL of the CA on which your router should send certificate requests; for example, http://ca_server. • *url*—Must be in the form of http://CA_name, where CA_name is the name of the CA's host Domain Name System or the IP address.
Step 3	Router(ca-trustpoint)# **subject-name** [*x.500-name*]	(Optional) Specifies the requested subject name that will be used in the certificate request. • *x.500-name*—If the *x.500-name* argument is not specified, the fully qualified domain name (FQDN), which is the default subject name, is used.

	Command	Purpose
Step 4	Router(ca-trustpoint)# **ip-address** {*interface* | **none**}	Includes the IP address of the specified interface in the certificate request. • *interface*—IP address of the interface. • **none**—Specify this keyword if no IP address should be included. If this command is enabled, you will not be prompted for an IP address during enrollment for this trustpoint.
Step 5	Router(ca-trustpoint)# **serial-number** [**none**]	Specifies the router serial number in the certificate request, unless the **none** keyword is issued. • **none**—(Optional) Specify this keyword if no serial number should be included.
Step 6	Router(ca-trustpoint)# **regenerate**	Enables key rollover with certificate enrollment when the **crypto pki enroll** command is issued. **Note** This command generates a new key for the certificate even if a named key already exists. Do not use the **crypto key generate** command with the key rollover feature. If the key pair being rolled over is exportable, the new key pair will also be exportable. The following comment will appear in the trustpoint configuration to indicate whether the key pair is exportable: `! RSA key pair associated with trustpoint is exportable.`
Step 7	Router(ca-trustpoint)# **password** *string*	(Optional) Specifies the revocation password for the certificate. • *string*—Text of the password. **Note** If this command is enabled, you will not be prompted for a password during enrollment for this trustpoint.

	Command	Purpose
Step 8	Router(ca-trustpoint)# **rsakeypair** *key-label* [*key-siz*e [*encryption-key-size*]]	Specifies which key pair to associate with the certificate. • *key-label*—Name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is configured. • *key-size*—(Optional) Size of the desired RSA key. If not specified, the existing key size is used. (The specified size must be the same as the *encryption-key-size*.) • *encryption-key-size*—(Optional) Size of the second key, which is used to request separate encryption, signature keys, and certificates. (The specified size must be the same as the *key-size*.) **Note** If this command is not enabled, the FQDN key pair is used.
Step 9	Router(ca-trustpoint)# **exit**	Exits ca-trustpoint configuration mode and enters global configuration mode.
Step 10	Router(config)# **crypto pki authenticate** *name*	Authenticates the CA (by obtaining the certificate of the CA.) • *name*—Name of the CA. Enter the *name* value entered in Step 1. Check the certificate fingerprint if prompted. **Note** This command is optional if the CA certificate is already loaded into the configuration.
Step 11	Router(config)# **crypto pki enroll** *name*	Requests certificates for all of your RSA key pairs. • *name*—Name of the CA. This command causes your router to request as many certificates as there are RSA key pairs, so you need perform this command only once, even if you have special-usage RSA key pairs. When the **regenerate** configuration command is configured, this command will perform key rollover. **Note** This command requires you to create a challenge password that is not saved with the configuration. This password is required if your certificate needs to be revoked, so you must remember this password.
Step 12	Router(config)# **exit**	Exits global configuration mode.

For complete configuration information for Key Rollover for Certificate Renewal, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/gtkyroll.htm

For key rollover configuration examples, see the "Key Rollover for Certificate Renewal Configuration Examples" section on page 26-59.

Configuring PKI: Query Multiple Servers During Certificate Revocation Check

Before an X.509 certificate presented by a peer is validated, the certificate revocation list (CRL) is checked to make sure that the certificate has not been revoked by the issuing certificate authority (CA). The certificate usually contains a certificate distribution point (CDP) in the form of a URL. Cisco IOS software uses the CDP to locate and retrieve the CRL.

Previous versions of Cisco IOS software make only one attempt to retrieve the CRL, even when the certificate contains more than one CDP. If the CDP server does not respond, the Cisco IOS software reports an error, which may result in the peer's certificate being rejected.

The PKI: Query Multiple Servers During Certificate Revocation Check feature provides the ability for Cisco IOS software to make multiple attempts to retrieve the CRL by trying all of the available CDPs in a certificate. This allows operations to continue when a particular server is not available. In addition, the ability to override the CDPs in a certificate with a manually configured CDP is also provided. Manually overriding the CDPs in a certificate can be advantageous when a particular server is unavailable for an extended period of time. The certificate's CDPs can be replaced with a URL or directory specification without reissuing all of the certificates that contain the original CDP.

To manually override the existing CDPs for a certificate with a URL or directory specification, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router (config)# **crypto pki trustpoint** *name*	Declares the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name for the trustpoint CA.
Step 2	Router(ca-trustpoint)# **match certificate** *certificate-map-label* **override cdp** {**url** | **directory**} *string*	Manually overrides the existing CDP entries for a certificate with a URL or directory specification. • *certificate-map-label*—A user-specified label that must match the label argument specified in a previously defined **crypto pki certificate map** command. • **url**—Specifies that the certificate's CDPs will be overridden with an HTTP or LDAP URL. • **directory**—Specifies that the certificate's CDPs will be overridden with an LDAP directory specification. • *string*—The URL or directory specification. Some applications may time out before all CDPs have been tried and will report an error message. This will not affect the router, and the Cisco IOS software will continue attempting to retrieve a CRL until all CDPs have been tried.

For complete configuration information for the PKI: Query Multiple Servers During Certificate Revocation Check feature, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/gtcertrc.htm

For a query multiple servers configuration example, see the "PKI: Query Multiple Servers During Certificate Revocation Check (CDP Override) Configuration Example" section on page 26-60.

Configuring the Online Certificate Status Protocol

The Online Certificate Status Protocol (OCSP) feature allows users to enable OCSP instead of certificate revocation lists (CRLs) to check certificate status. Unlike CRLs, which provide only periodic certificate status, OCSP can provide timely information regarding the status of a certificate.

OSCP Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring OCSP:

- OCSP transports messages over HTTP, so there may be a time delay when you access the OCSP server. If the OCSP server is unavailable, certificate verification will fail.
- The increased certificate size may cause a problem for low-end routers when certificates are stored on NVRAM. Thus, before you add the Authority Info Access (AIA) extension to a certificate, make sure that the increased size will not cause deployment problems.

- An OCSP server usually operates in either push or poll mode. You can configure a CA server to push revocation information to an OCSP server or configure an OCSP server to periodically download (poll) a CRL from the CA server. To ensure that timely certificate revocation status is obtained, you should carefully consider the “push and poll” interval.
- When configuring an OCSP server to return the revocation status for a CA server, the OCSP server must be configured with an OCSP response signing certificate that is issued by that CA server. Ensure that the signing certificate is in the correct format, or the router will not accept the OCSP response. Refer to your OCSP manual for additional information.

To configure your router for OCSP to check certificate status, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the CA that your router should use and puts you in ca-trustpoint configuration mode. • *name*—Name for the trustpoint CA.
Step 2	Router(ca-trustpoint)# **ocsp url** *url*	(Optional) Specifies the URL of an OCSP server so that the trustpoint can check the certificate status. This URL will override the URL of the OCSP server (if one exists) in the Authority Info Access (AIA) extension of the certificate. • *url* —Specifies the HTTP URL to be used.
Step 3	Router(ca-trustpoint)# **revocation-check** *method1* [*method2*[*method3*]]	Checks the revocation status of a certificate. • *method1* [*method2*[*method3*]]—Specifies the method used by the router to check the revocation status of the certificate. Available methods are as follows: – **crl**—Certificate checking is performed by a CRL. This is the default option. – **none**—Certificate checking is ignored. – **ocsp**—Certificate checking is performed by an OCSP server. If a second and third method are specified, each method will be used only if the previous method returns an error, such as a server being down.

Verifying the OCSP Configuration

To display information about your certificate and the CA certificate, enter the **show crypto pki certificates** command:

```
Router# show crypto pki certificates

Certificate

  Status: Available
  Version: 3
  Certificate Serial Number: 18C1EE03000000004CBD
  Certificate Usage: General Purpose
```

```
Issuer:
  cn=msca-root
  ou=pki msca-root
  o=cisco
  l=santa cruz2
  st=CA
  c=US
  ea=user@example.com

Subject:
  Name: myrouter.example.com
  hostname=myrouter.example.com

CRL Distribution Points:
  http://msca-root/CertEnroll/msca-root.crl

Validity Date:
  start date: 19:50:40 GMT Oct 5 2004
  end   date: 20:00:40 GMT Oct 12 2004

  Subject Key Info:
  Public Key Algorithm: rsaEncryption
  RSA Public Key: (360 bit)

  Signature Algorithm: SHA1 with RSA Encryption
  Fingerprint MD5: 2B5F53E6 E3E892E6 3A9D3706 01261F10
  Fingerprint SHA1: 315D127C 3AD34010 40CE7F3A 988BBDA5 CD528824

  X509v3 extensions:
  X509v3 Key Usage: A0000000
  Digital Signature
  Key Encipherment
  X509v3 Subject Key ID: D156E92F 46739CBA DFE66D2D 3559483E B41ECCF4
  X509v3 Authority Key ID: 37F3CC61 AF5E7C0B 434AB364 CF9FA0C1 B17C50D9
  Authority Info Access:

  Associated Trustpoints: msca-root
  Key Label: myrouter.example.com

 CA Certificate

  Status: Available
  Version: 3
  Certificate Serial Number: 1244325DE0369880465F977A18F61CA8
  Certificate Usage: Signature

  Issuer:
  cn=msca-root
  ou=pki msca-root
  o=cisco
  l=santa cruz2
  st=CA
  c=US
  ea=user@example.com

  Subject:
  cn=msca-root
  ou=pki msca-root
  o=cisco
  l=santa cruz2
  st=CA
  c=US
  ea=user@example.com
```

```
    CRL Distribution Points:
    http://msca-root.example.com/CertEnroll/msca-root.crl

    Validity Date:
    start date: 22:19:29 GMT Oct 31 2002
    end   date: 22:27:27 GMT Oct 31 2017

    Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (512 bit)

    Signature Algorithm: SHA1 with RSA Encryption
    Fingerprint MD5: 84E470A2 38176CB1 AA0476B9 C0B4F478
    Fingerprint SHA1: 0F57170C 654A5D7D 10973553 EFB0F94F 2FAF9837

    X509v3 extensions:
    X509v3 Key Usage: C6000000
    Digital Signature
    Non Repudiation
    Key Cert Sign
    CRL Signature

  X509v3 Subject Key ID: 37F3CC61 AF5E7C0B 434AB364 CF9FA0C1 B17C50D9
  X509v3 Basic Constraints:
    CA: TRUE

  Authority Info Access:
  Associated Trustpoints: msca-root
```

To display the trustpoints and configured trustpoint subcommands that are configured in the router, enter the **show crypto pki trustpoints** command:

```
Router# show crypto pki trustpoints

Trustpoint bo:
    Subject Name:
    CN = bomborra Certificate Manager
    O = cisco.com
    C = US
    Serial Number:01
    Certificate configured.
    CEP URL:http://bomborra
    CRL query url:ldap://bomborra
```

For complete configuration information for OCSP, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_2/gt_ocsp.htm

For OCSP configuration examples, see the "Online Certificate Status Protocol Configuration Examples" section on page 26-60.

Configuring Optional OCSP Nonces

The Optional OCSP Nonces feature provides users with the ability to disable the sending of a nonce, or unique identifier for an Online Certificate Status Protocol (OCSP) request, during OCSP communications.

Note The Optional OCSP Nonces feature is only supported as of Cisco IOS Release 12.2(33)SRA.

When using OCSP as your revocation method, unique identifiers, or nonces, are sent by default during peer communications with the OCSP server. The use of unique identifiers during OCSP server communications enables more secure and reliable communications. However, not all OCSP servers support the use of unique identifiers. (Refer to your OCSP manual for more information.) To disable the use of unique identifiers during OCSP communications, use the **ocsp disable-nonce** subcommand in the **crypto pki trustpoint** command.

Disabling OCSP Nonces

By default, OCSP nonces are used. To disable the use of these nonces and specify that your router should not send unique identifiers, or nonces, during OCSP communication, use the **ocsp disable-nonce** subcommand in the **crypto pki trustpoint** command as follows beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the certificate authority (CA) that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 2	Router (ca-trustpoint)# **ocsp disable-nonce**	Specifies that your router will not send unique identifiers, or nonces, during OCSP communications.
Step 3	Router(ca-trustpoint)# **end**	(Optional) Exits ca-trustpoint configuration mode.

For complete configuration information for optional OCSP nonces, refer to this URL:

http://www.cisco.com/en/US/products/ps6441/products_configuration_guide_chapter09186a008051eaf8.html

For an optional OCSP nonces configuration example, see the "Optional OCSP Nonces Configuration Example" section on page 26-61.

Configuring Certificate Security Attribute-Based Access Control

Under the IPSec protocol, Certificate Authority (CA) interoperability permits Cisco IOS devices and a CA to communicate so that the Cisco IOS device can obtain and use digital certificates from the CA. Certificates contain several fields that are used to determine whether a device or user is authorized to perform a specified action. The Certificate Security Attribute-Based Access Control feature adds fields to the certificate to create a certificate-based ACL.

Certificate Security Attribute-Based Access Control Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Certificate Security Attribute-Based Access Control:

- The certificate-based ACL specifies one or more fields within the certificate and an acceptable value for each specified field. You can specify which fields within a certificate should be checked and which values those fields may or may not have. There are six logical tests for comparing the field with the value—equal, not equal, contains, does not contain, less than, and greater than or equal.
- If more than one field is specified within a single certificate-based ACL, the tests of all of the fields within the ACL must succeed to match the ACL.
- The same field can be specified multiple times within the same ACL.
- More than one ACL can be specified. Each ACL will be processed in turn until a match is found or all of the ACLs have been processed.
- Memory is required to hold the ACLs as they are created and as they are loaded from the configuration file. The amount of memory depends on which fields within the certificate are being checked and how many ACLs have been defined. Certificate-based ACL support requires one or more compare operations when the fields in a certificate are being checked. Only the fields specified by the ACL are checked. The compare operations are a small part of certificate validation and will not have a noticeable effect on router performance when validating certificates.

To configure Certificate Security Attribute-Based Access Control, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki certificate map** *label sequence-number*	Starts ca-certificate-map mode and defines certificate-based ACLs by assigning a label for the ACL that will also be referenced within the **crypto pki trustpoint** command. • *label*—An arbitrary string that identifies the ACL. • *sequence-number*—A sequence number that orders ACLs with the same label.
Step 2	Router(ca-certificate-map)# *field-name match-criteria match-value*	In ca-certificate-map mode, you specify one or more certificate fields together with their matching criteria and the value to match. • *field-name*—Specifies one of the following case-insensitive name strings or a date: – **subject-name** – **issuer-name** – **unstructured-subject-name** – **alt-subject-name** – **name** – **valid-start** – **expires-on** **Note** Date field format is *dd mm yyyy hh*:*mm*:*ss* or *mmm dd yyyy hh*:*mm*:*ss*. • *match-criteria*—Specifies one of the following logical operators: – **eq**—equal (valid for name and date fields) – **ne**—not equal (valid for name and date fields) – **co**—contains (valid only for name fields) – **nc**—does not contain (valid only for name fields) – **lt** —less than (valid only for date fields) – **ge** —greater than or equal (valid only for date fields) • *match-value*—Specifies the name or date to test with the logical operator assigned by *match-criteria*. For example: `Router(ca-certificate-map)# subject-name co Cisco`

	Command	Purpose
Step 3	Router(ca-certificate-map)# **exit**	Exits ca-certificate-map mode.
Step 4	Router(config)# **crypto pki trustpoint** *name*	Starts ca-trustpoint configuration mode and creates a name for the CA. • *name*—Specifies a name for the CA.
Step 5	Router(ca-trustpoint)# **match certificate** *certificate-map-label*	Associates the certificate-based ACL defined with the **crypto pki certificate map** command to the trustpoint. • *certificate-map-label*—Specifies the label argument specified in the previously defined **crypto pki certificate map** command in Step 1.
Step 6	Router(ca-trustpoint)# **exit**	Exits ca-trustpoint configuration mode.

Verifying Certificate-Based ACLs

To verify the certificate-based ACL configuration, enter the **show crypto pki certificates** command. The following example shows the components of the certificates—CA and router certificate—installed on the router when the router has both authenticated and enrolled with a trustpoint:

```
Router# show crypto pki certificates

CA Certificate
   Status: Available
   Certificate Serial Number: 1244325DE0369880465F977A18F61CA8
   Certificate Usage: Signature

   Issuer:
    CN = new-user
    OU = pki new-user
    O = cisco
    L = santa cruz2
    ST = CA
    C = US
    EA = user@cysco.net

    Subject:
    CN = new-user
    OU = pki new-user
    O = cisco
    L = santa cruz2
    ST = CA
    C = US
    EA = user@cysco.net

    CRL Distribution Point:
    http://new-user.cysco.net/CertEnroll/new-user.crl

    Validity Date:
    start date: 14:19:29 PST Oct 31 2002
    end date: 14:27:27 PST Oct 31 2017

    Associated Trustpoints: MS


Certificate
  Status: Available
```

```
Certificate Serial Number: 193E28D20000000009F7
Certificate Usage: Signature

Issuer:
   CN = new-user
   OU = pki new-user
   O = cisco
   L = santa cruz2
   ST = CA
  C = US
   EA = user@cysco.net

Subject:
  Name: User1.Cysco.Net
  OID.1.2.840.113549.1.9.2 = User1.Cysco.Net

CRL Distribution Point:
  http://new-user.cysco.net/CertEnroll/new-user.crl

Validity Date:
  start date: 12:40:14 PST Feb 26 2003
  end   date: 12:50:14 PST Mar 5 2003
  renew date: 16:00:00 PST Dec 31 1969

Associated Trustpoints: MS
```

For complete configuration information for Certificate Security Attribute-Based Access Control, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t15/ftcrtacl.htm

For a certificate-based ACL example, see the "Certificate Security Attribute-Based Access Control Configuration Example" section on page 26-61.

Configuring PKI AAA Authorization Using the Entire Subject Name

When using public key infrastructure (PKI) and authentication, authorization, and accounting (AAA) functionality, users sometimes have attribute-value (AV) pairs that are different from those of every other user. As a result, a unique username is required for each user. The PKI AAA Authorization Using the Entire Subject Name feature provides users with the ability to query the AAA server using the entire subject name from the certificate as a unique AAA username.

PKI AAA Authorization Using the Entire Subject Name Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring PKI AAA Authorization Using the Entire Subject Name:

- Some AAA servers limit the length of the username (for example, to 64 characters). As a result, the entire certificate subject name cannot be longer than the limitation of the server.
- Some AAA servers limit the available character set that may be used for the username (for example, a space [ ] and an equal sign [=] may not be acceptable). This feature will not work for the AAA server having such a character-set limitation.

- The **subject-name** command in the trustpoint configuration might not always be the final AAA subject name. If the fully qualified domain name (FQDN), serial number, or IP address of the router are included in a certificate request, the subject name field of the issued certificate will also have these components. To turn off the components, use the **fqdn**, **serial-number**, and **ip-address** commands with the **none** keyword.
- Certificate authority (CA) servers sometimes change the requested subject name field when they issue a certificate. For example, CA servers of some vendors switch the relative distinguished names (RDNs) in the requested subject names to the following order: CN, OU, O, L, ST, and C. However, another CA server might append the configured Lightweight Directory Access Protocol (LDAP) directory root (for example, O=cisco.com) to the end of the requested subject name.
- Depending on the tools you choose for displaying a certificate, the printed order of the RDNs in the subject name could be different. Cisco IOS software always displays the least significant RDN first, but other software, such as Open Source Secure Socket Layer (OpenSSL), does the opposite. Therefore, if you are configuring the AAA server with a full DN (subject name) as the corresponding username, ensure that the Cisco IOS software style (that is, with the least-significant RDN first) is used.

To configure the entire certificate subject name for PKI authentication, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **aaa new-model**	Enables the AAA access control model.
Step 2	Router config)# **aaa authorization network** *listname* [*method*]	Sets the parameters that restrict user access to a network. • *listname*—Character string used to name the list of authorization methods. • *method*—(Optional) Specifies an authorization method to be used for authorization. The *method* argument can be **group radius**, **group tacacs+**, or **group** *group-name*.
Step 3	Router(config)# **crypto pki trustpoint** *name*	Declares the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 4	Router(ca-trustpoint)# **enrollment url** *url*	Specifies the enrollment parameters of your CA. • *url*—The *url* argument is the URL of the CA to which your router should send certificate requests.
Step 5	Router(ca-trustpoint)# **revocation-check** *method*	(Optional) Checks the revocation status of a certificate. • *method*—Method used by the router to check the revocation status. Available methods are **ocsp**, **none**, and **crl**.
Step 6	Router(ca-trustpoint)# **exit**	Exits ca-truspoint configuration mode and enters global configuration mode.
Step 7	Router(config)# **authorization list** {*listname*}	Specifies the AAA authorization list. • *listname*—Name of the list.

	Command	Purpose
Step 8	Router(config)# **authorization username subjectname all**	Sets parameters for the different certificate fields that are used to build the AAA username. The **all** parameter specifies that the entire subject name of the certificate will be used as the authorization username.
Step 9	Router(config)# **tacacs-server host** *hostname* [**key** *string*] or Router (config)# **radius-server host** *hostname* [**key** *string*]	Specifies a TACACS+ host. • *hostname*—Name of the host. • **key** *string*—(Optional) Character string specifying authentication and encryption key. or Specifies a RADIUS host.

For complete configuration information for the PKI AAA Authorization Using the Entire Subject Name feature, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_11/gt_dnall.htm

For a PKI AAA Authorization Using the Entire Subject Name configuration example, see the "PKI AAA Authentication Using the Entire Subject Name Configuration Example" section on page 26-62.

Configuring Source Interface Selection for Outgoing Traffic with Certificate Authority

The Source Interface Selection for Outgoing Traffic with Certificate Authority feature allows you to specify that the address of an interface be used as the source address for all outgoing TCP connections associated with that trustpoint when a designated trustpoint has been configured.

To configure the interface that you want to use as the source address for all outgoing TCP connections associated with a trustpoint, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoin**t *name*	Declares the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Name for the trustpoint CA.
Step 2	Router(ca-trustpoint)# **enrollment url** *url*	Specifies the enrollment parameters of your CA. • *url*—Specifies the URL of the CA where your router should send certificate requests; for example, http://ca_server. *url* must be in the form http://CA_*name*, where CA_*name* is the CA's host Domain Name System (DNS) name or IP address.

	Command	Purpose
Step 3	Router(ca-trustpoint)# **source interface** *interface-address*	Specifies the interface to be used as the source address for all outgoing TCP connections associated with that trustpoint. • *interface-address*—Interface address.
Step 4	Router(config)# **interface** *type slot*/[*subslot*]/*port*	Configures an interface type and enters interface configuration mode. • *type*—Type of interface being configured. • *slot*/[*subslot*]/ *port*—Number of the slot, subslot (optional), and port to be configured.
Step 5	Router(config-if)# **description** *string*	Adds a description to an interface configuration. • *string*—Descriptive string.
Step 6	Router(config-if)# **ip address** *ip-address mask*	Sets a primary or secondary IP address for an interface. • *address*—IP address. • *mask*—Subnet mask.
Step 7	Router(config-if)# **interface** *type slot*/[*subslot*]/*port*	Configures an interface type. • *type*—Type of interface being configured. • *slot*/[*subslot*]/ *port*—Number of the slot, subslot (optional), and port to be configured.
Step 8	Router(config-if)# **description** *string*	Adds a description to an interface configuration. • *string*—Descriptive string.
Step 9	Router(config-if)# **ip address** *ip-address mask* [*secondary*]	Sets a primary or secondary IP address for an interface. • *address*—IP address. • *mask*—Subnet mask. • [*secondary*]—(Optional) Secondary address.
Step 10	Router(config-if)# **crypto map** *map-name*	Applies a previously defined crypto map set to the interface. • *map-name*—Name that identifies the crypto map set.

For complete configuration information for Source Interface Selection for Outgoing Traffic with Certificate Authority, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t15/ft_asish.htm

For a source interface selection configuration example, see the "Source Interface Selection for Outgoing Traffic with Certificate Authority Configuration Example" section on page 26-62.

Configuring Persistent Self-Signed Certificates

The Persistent Self-Signed Certificates feature saves a certificate generated by a Secure HTTP (HTTPS) server for the Secure Sockets Layer (SSL) handshake in a router's startup configuration.

Note The Persistent Self-Signed Certificates feature is only supported as of Cisco IOS Release 12.2(33)SRA.

Cisco IOS software has an HTTPS server that allows access to web-based management pages using a secure SSL connection. SSL requires the server to have an X.509 certificate that is sent to the client (web browser) during the SSL handshake to establish a secure connection between the server and the client.

The client expects the SSL server's certificate to be verifiable using a certificate the client already possesses.

If Cisco IOS software does not have a certificate that the HTTPS server can use, the server generates a self-signed certificate by calling a public key infrastructure (PKI) application programming interface (API). When the client receives this self-signed certificate and is unable to verify it, intervention is needed. The client asks you if the certificate should be accepted and saved for future use. If you accept the certificate, the SSL handshake continues.

Future SSL handshakes between the same client and the server use the same certificate. However, if the router is reloaded, the self-signed certificate is lost. The HTTPS server must then create a new self-signed certificate. This new self-signed certificate does not match the previous certificate, so you are once again asked to accept it.

Requesting acceptance of the router's certificate each time that the router reloads can be annoying and may present an opportunity for an attacker to substitute an unauthorized certificate during the time that you are being asked to accept the certificate.

The Persistent Self-Signed Certificates feature overcomes all these limitations by saving a certificate in the router's startup configuration, resulting in the following benefits:

- Having a persistent self-signed certificate stored in the router's startup configuration (NVRAM) lessens the opportunity for an attacker to substitute an unauthorized certificate because the browser is able to compare the certificate offered by the router with the previously saved certificate and warn you if the certificate has changed.
- Having a persistent self-signed certificate stored in the router's startup configuration eliminates the user intervention that is necessary to accept the certificate every time that the router reloads.
- Because user intervention is no longer necessary to accept the certificate, the secure connection process is faster.

Persistent Self-Signed Certificates Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring persistent self-signed certificates:

- You must load an image that supports SSL.
- You can configure only one trustpoint for a persistent self-signed certificate.

Configuring a Trustpoint and Specifying Self-Signed Certificate Parameters

To configure a trustpoint and specify self-signed certificate parameters, perform the following steps beginning in global configuration mode:

Note This section is optional because if you enable the Secure HTTP (HTTPS) server, it generates a self-signed certificate automatically using default values. To specify parameters, you must create a trustpoint and configure it. To use default values, delete any existing self-signed trustpoints. Deleting all self-signed trustpoints causes the HTTPS server to generate a persistent self-signed certificate using default values as soon as it is enabled.

	Command	Purpose
Step 1	Router(config)# **crypto pki trustpoint** *name*	Declares the certificate authority (CA) that your router should use and enters ca-trustpoint configuration mode. • *name*—Name of the CA.
Step 2	Router(ca-trustpoint)# **enrollment selfsigned**	Specifies self-signed enrollment.
Step 3	Router(ca-trustpoint)# **subject-name** [*x.500-name*]	(Optional) Specifies the requested subject name to be used in the certificate request. • *x.500-name*—If the x.500-name argument is not specified, the fully qualified domain name (FQDN), which is the default subject name, is used.
Step 4	Router(ca-trustpoint)# **rsakeypair** *key-label* [*key-size* [*encryption-key-size*]]	(Optional) Specifies which key pair to associate with the certificate. • *key-label*—Name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is configured. • *key-size*—(Optional) Size of the desired RSA key. If not specified, the existing key size is used. (The specified size must be the same as the *encryption-key-size*.) • *encryption-key-size*—(Optional) Size of the second key, which is used to request separate encryption, signature keys, and certificates. (The specified size must be the same as the *key-size*.) **Note** If this command is not enabled, the FQDN key pair is used.
Step 5	Router(ca-trustpoint)# **crypto pki enroll** *trustpoint-name*	Tells the router to generate the persistent self-signed certificate. • *trustpoint-name*—Name of the CA.
Step 6	Router(ca-trustpoint)# **end**	(Optional) Exits ca-trustpoint configuration mode.

Enabling the HTTPS Server

To enable the HTTPS server, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **ip http secure-server**	Enables the secure HTTP web server. **Note** A key pair (modulus 1024) and a certificate are generated.
Step 2	Router(config)# **end**	Exits global configuration mode.

Note You must issue a **write memory** command to save the configuration. This command also saves the self-signed certificate and the HTTPS server in enabled mode.

Verifying the Persistent Self-Signed Certificate Configuration

To verify that a self-signed certificate and a trustpoint have been created, enter the **show crypto pki certificates**, **show crypto mypubkey rsa,** and the **show crypto pki trustpoints** commands:

The **show crypto pki certificates** command displays information about your certificate, the CA certificate, and any registration authority certificates:

```
Router# show crypto pki certificates

Router Self-Signed Certificate
   Status: Available
   Certificate Serial Number: 01
   Certificate Usage: General Purpose
   Issuer:
    cn=IOS-Self-Signed-Certificate-3326000105
   Subject:
    Name: IOS-Self-Signed-Certificate-3326000105
    cn=IOS-Self-Signed-Certificate-3326000105
   Validity Date:
    start date: 19:14:14 GMT Dec 21 2004
    end   date: 00:00:00 GMT Jan 1 2020
   Associated Trustpoints: TP-self-signed-3326000105
```

Note The number 3326000105 above is the router's serial number and varies depending on the router's actual serial number.

The **show crypto mypubkey rsa** command displays information about the key pair corresponding to the self-signed certificate:

```
Router# show crypto mypubkey rsa

% Key pair was generated at: 19:14:10 GMT Dec 21 2004
Key name: TP-self-signed-3326000105
  Usage: General Purpose Key
  Key is not exportable.
  Key Data:
    30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00B88F70
    6BC78B6D 67D6CFF3 135C1D91 8F360292 CA44A032 5AC1A8FD 095E4865 F8C95A2B
    BFD1C2B7 E64A3804 9BBD7326 207BD456 19BAB78B D075E78E 00D2560C B09289AE
```

```
    6DECB8B0 6672FB3A 5CDAEE92 9D4C4F71 F3BCB269 214F6293 4BA8FABF 9486BCFC
    2B941BCA 550999A7 2EFE12A5 6B7B669A 2D88AB77 39B38E0E AA23CB8C B7020301 0001
% Key pair was generated at: 19:14:13 GMT Dec 21 2004
Key name: TP-self-signed-3326000105.server
  Usage: Encryption Key
  Key is not exportable.
  Key Data:
    307C300D 06092A86 4886F70D 01010105 00036B00 30680261 00C5680E 89777B42
    463E5783 FE96EA9E F446DC7B 70499AF3 EA266651 56EE29F4 5B003D93 2FC9F81D
    8A46E12F 3FBAC2F3 046ED9DD C5F27C20 1BBA6B9B 08F16E45 C34D6337 F863D605
    34E30F0E B4921BC5 DAC9EBBA 50C54AA0 BF551BDD 88453F50 61020301 0001
```

Note The second key pair with the name TP-self-signed-3326000105.server is the SSH key pair and is generated once any key pair is created on the router and SSH starts up.

The **show crypto pki trustpoints** command displays the trustpoints that are configured in the router:

```
Router# show crypto pki trustpoints

Trustpoint local:
    Subject Name:
    serialNumber=C63EBBE9+ipaddress=10.3.0.18+hostname=test.cisco.com
        Serial Number: 01
    Persistent self-signed certificate trust point
```

For complete configuration information for persistent self-signed certificates, refer to this URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a008040adf0.html

For persistent self-signed certificates configuration examples, see the "Persistent Self-Signed Certificates Configuration Examples" section on page 26-63.

Configuring Certificate Chain Verification

To determine if a trustpoint has been successfully authenticated, a certificate has been requested and granted, and if the certificate is currently valid, enter the **crypto pki cert validate** command.

Note The **crypto pki cert validate** command is only supported as of Cisco IOS Release 12.2(33)SRA.

Certificate Chain Verification Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring certificate chain verification:

- The **crypto pki cert validate** command validates the router's own certificate for a given trustpoint. Use this command after enrollment to verify that the trustpoint is properly authenticated, a certificate has been requested and granted for the trustpoint, and that the certificate is currently valid. A certificate is valid if it is signed by the trustpoint certificate authority (CA), not expired, and so on.

To allow the router to send dead peer detection (DPD) messages to the peer, enter the **crypto pki cert validate** command in global configuration mode as follows:

Router(config)# **crypto pki cert validate** *trustpoint*

In this command, *trustpoint* specifies the trustpoint to be validated.

For complete configuration information for certificate chain verification, refer to the *Cisco IOS Security Command Reference*.

For certificate chain verification configuration examples, see the "Certificate Chain Verification Configuration Examples" section on page 26-64.

Configuration Examples

This section provides examples of the following configurations:

RSA Key Pairs Configuration Example

The following example is a sample trustpoint configuration that specifies the RSA key pair "exampleCAkeys":

```
Router(config)# crypto key generate rsa general-purpose exampleCAkeys
Router(config)# crypto pki trustpoint exampleCAkeys
Router(config)# enroll url http://exampleCAkeys/certsrv/mscep/mscep.dll
Router(config)# rsakeypair exampleCAkeys 1024 1024
```

Protected Private Key Storage Configuration Examples

This section contains the following configuration examples:

- Encrypted Key Configuration Example, page 26-53
- Locked Key Configuration Example, page 26-53

Encrypted Key Configuration Example

The following example shows how to encrypt the RSA key "pki1-72a.cisco.com":

```
Router(config)# crypto key encrypt rsa name pki1-72a.cisco.com passphrase cisco1234
Router(config)# exit
```

Locked Key Configuration Example

The following example shows how to lock the key "pki1-72a.cisco.com":

```
Router# crypto key lock rsa name pki1-72a.cisco.com passphrase cisco1234
```

Trustpoint CA Configuration Example

The following example shows how to declare the CA named "kahului" and specify characteristics for the trustpoint CA:

```
Router(config)# crypto pki trustpoint kahului
Router(ca-trustpoint)# enrollment url http://kahului
Router(ca-trustpoint)# crl query ldap://kahului
```

Query Mode Definition Per Trustpoint Configuration Example

The following configuration example shows a trustpoint CA that uses query mode:

```
Router(config)# crypto pki trustpoint trustpoint1
Router(ca-trustpoint)# enrollment url http://ca-server1
Router(ca-trustpoint)#crl query http://ca-server1
Router(ca-trustpoint)# default query certificate
Router(ca-trustpoint)# query certificate
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate trustpoint1
Router(config)# crypto key generate rsa
Router(config)# crypto pki enroll trustpoint1
```

Local Certificate Storage Location Configuration Example

The following example shows how to store certificates to the certs subdirectory. Note that the certs subdirectory does not exist and is automatically created.

```
Router(config)# crypto pki certificate storage disk0:/certs
Requested directory does not exist -- created
Certificates will be stored in disk0:/certs/
Router(config)# end
Router# write
*May 27 02:09:00:%SYS-5-CONFIG_I:Configured from console by consolemem
Building configuration...
[OK]
Router# directory disk0:/certs
Directory of disk0:/certs/
14  -rw-         707  May 27 2005 02:09:02 +00:00  ioscaroot#7401CA.cer
15  -rw-         863  May 27 2005 02:09:02 +00:00  msca-root#826E.cer
16  -rw-         759  May 27 2005 02:09:02 +00:00  msca-root#1BA8CA.cer
17  -rw-         863  May 27 2005 02:09:02 +00:00  msca-root#75B8.cer
18  -rw-        1149  May 27 2005 02:09:02 +00:00  storagename#6500CA.cer
19  -rw-         863  May 27 2005 02:09:02 +00:00  msca-root#83EE.cer
47894528 bytes total (20934656 bytes free)

! The certificate files are now on disk0/certs:
```

Direct HTTP Enrollment with CA Servers Configuration Examples

This section provides the following configuration examples:

- Enrollment Profile for a Client Router Configuration Example, page 26-54
- Enrollment Profile for a Client Router Already Enrolled with a Third-Party Vendor CA Example, page 26-55
- Certificate Server Automatically Accepting Enrollment Requests Only from the Client Router Configuration Example, page 26-55

Enrollment Profile for a Client Router Configuration Example

The following example show how to configure an enrollment profile for direct HTTP enrollment with a CA server:

```
Router(config)# crypto pki trustpoint Entrust
Router(ca-trustpoint)# enrollment profile E
Router(ca-trustpoint)# exit
Router(config)# crypto pki profile enrollment E
Router(ca-profile-enroll)# authentication url  http://entrust:81
Router(ca-profile-enroll)# authentication command  GET /certs/cacert.der
Router(ca-profile-enroll)# enrollment url  http://entrust:81/cda-cgi/clientcgi.exe
Router(ca-profile-enroll)# enrollment command  POST reference_number=$P2&authcode=$P1
&retrievedAs=rawDER&action=getServerCert&pkcs10Request=$REQ
Router(ca-profile-enroll)# parameter 1 value aaaa-bbbb-cccc
Router(ca-profile-enroll)# parameter 2 value 5001
```

Enrollment Profile for a Client Router Already Enrolled with a Third-Party Vendor CA Example

The following example shows how to configure the following tasks on the client router:

- Define the trustpoint "msca-root" that points to the third-party vendor CA and enroll and authenticate the client with the third-party vendor CA.
- Define trustpoint "cs" for the Cisco IOS CA.
- Define enrollment profile "cs1," which points to Cisco IOS CA and mention (via the enrollment credential command) that "msca-root" is being initially enrolled with the Cisco IOS CA.

```
! Define trustpoint "msca-root" for non-Cisco IOS CA.
Router(config)# crypto pki trustpoint msca-root
Router(ca-trustpoint)# enrollment mode ra
Router(ca-trustpoint)# enrollment url http://msca-root:80/certsrv/mscep/mscep.dll
Router(ca-trustpoint)# ip-address FastEthernet2/0
Router(ca-trustpoint)#r evocation-check crl

! Configure trustpoint "cs" for Cisco IOS CA.
Router(config)# crypto pki trustpoint cs
Router(ca-trustpoint)# enrollment profile cs1
Router(ca-trustpoint)# revocation-check crl

! Define enrollment profile "cs1."
Router(config)# crypto pki profile enrollment cs1
Router(ca-profile-enroll)# enrollment url  http://cs:80
Router(ca-profile-enroll)# enrollment credential  msca-root
```

Certificate Server Automatically Accepting Enrollment Requests Only from the Client Router Configuration Example

The following example shows how to configure the certificate server, and issue the **grant auto trustpoint** command to instruct the certificate server to accept enrollment requests only from clients who are already enrolled with trustpoint "msca-root":

```
Router(config)# crypto pki server cs
Router(cs-server)# database level minimum
Router(cs-server)# database url nvram:
Router(cs-server)# issuer-name CN=cs
Router(cs-server)# grant auto trustpoint msca-root

Router(config)# crypto pki trustpoint cs
Router(ca-trustpoint)# revocation-check crl
Router(ca-trustpoint)# rsakeypair cs

Router(ca-trustpoint)# crypto pki trustpoint msca-root
Router(ca-trustpoint)# enrollment mode ra
Router(ca-trustpoint)# enrollment url http://msca-root:80/certsrv/mscep/mscep.dll
Router(ca-trustpoint)# revocation-check crl
```

Manual Certificate Enrollment Configuration Examples

This section provides the following manual certificate enrollment configuration examples:

- Manual Certificate Enrollment Using TFTP Configuration Example, page 26-56
- Manual Certificate Enrollment Using Cut-and-Paste Configuration Example, page 26-56

Manual Certificate Enrollment Using TFTP Configuration Example

The following example shows the configuration of manual certificate enrollment using TFTP:

```
Router(config)# crypto pki trustpoint MS
Router(ca-trustpoint)# enrollment url tftp://CA-Server/TFTPfiles/router1
Router(ca-trustpoint)# crypto pki authenticate MS
Router(ca-trustpoint)# exit
Router(config)# crypto pki enroll MS
Router(config)# crypto pki import MS certificate
```

Manual Certificate Enrollment Using Cut-and-Paste Configuration Example

The following example shows how to configure manual cut-and-paste certificate enrollment. In this example, the name of the trustpoint CA is "MS," and the **crypto pki import** command is entered twice because usage keys (signature and encryption keys) are used.

```
Router(config)# crypto pki trustpoint MS
Router(ca-trustpoint)# enrollment terminal
Router(ca-trustpoint)# crypto pki authenticate MS

Enter the base 64 encoded CA certificate.

End with a blank line or the word "quit" on a line by itself

-----BEGIN CERTIFICATE-----

MIICNDCCAd6gAwIBAgIQOsCmXpVHwodKryRoqULV7jANBgkqhkiG9w0BAQUFADA5
MQswCQYDVQQGEwJVUzEWMBQGA1UEChMNQ2lzY28gU3lzdGVtczESMBAGA1UEAxMJ
bXNjYS1yb290MB4XDTAyMDIxNDAwNDYwMVoXDTA3MDIxNDAwNTQ0OFowOTELMAkG
A1UEBhMCVVMxFjAUBgNVBAoTDUNpc2NvIFN5c3RlbXMxEjAQBgNVBAMTCW1zY2Et
cm9vdDBcMA0GCSqGSIb3DQEBAQUAA0sAMEgCQQCix8nIGFg+wvy3BjFbVi25wYoG
K2N0HWWHpqxFuFhqyBnIC0OshIn9CtrdN3JvUNHr0NIKocEwNKUGYmPwWGTfAgMB
AAGjgcEwgb4wCwYDVR0PBAQDAgHGMA8GA1UdEwEB/wQFMAMBAf8wHQYDVR0OBBYE
FKIacsl6dKAfuNDVQymlSp7esf8jMG0GA1UdHwRmMGQwL6AtoCuGKWh0dHA6Ly9t
c2NhLXJvb3QvQ2VydEVucm9sbC9tc2NhLXJvb3QuY3JsMDGgL6AthitmaWxlOi8v
XFxtc2NhLXJvb3RcQ2VydEVucm9sbFxtc2NhLXJvb3QuY3JsMBAGCSsGAQQBgjcV
AQQDAgEAMA0GCSqGSIb3DQEBBQUAA0EAeuZkZMX9qkoLHfETYTpVWjZPQbBmwNRA
oJDSdYdtL3BcI/uLL5q7EmODyGfLyMGxuhQYx5r/40aSQgLCqBq+yg==

-----END CERTIFICATE-----

Certificate has the following attributes:

Fingerprint:D6C12961 CD78808A 4E02193C 0790082A

% Do you accept this certificate? [yes/no]:y

Trustpoint CA certificate accepted.

% Certificate successfully imported

Router(config)#
```

```
Router(config)# crypto pki enroll MS

% Start certificate enrollment..

% The subject name in the certificate will be:Router.cisco.com

% Include the router serial number in the subject name? [yes/no]:n

% Include an IP address in the subject name? [no]:n

Display Certificate Request to terminal? [yes/no]:y

Signature key certificate request -

Certificate Request follows:

MIIBhTCB7wIBADAlMSMwIQYJKoZIhvcNAQkCFhRTYW5kQmFnZ2VyLmNpc2NvLmNv
bTCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAxdhXFDiWAn/hIZs9zfOtssKA
daoWYu0ms9Fe/Pew01dh14vXdxgacstOs2Pr5wk6jLOPxpvxOJPWyQM6ipLmyVxv
ojhyLTrVohrh6Dnqcvk+G/5ohss9o9RxvONwx042pQchFnx9EkMuZC7evwRxJEqR
mBHXBZ8GmP3jYQsjS8MCAwEAAaAhMB8GCSqGSIb3DQEJDjESMBAwDgYDVR0PAQH/
BAQDAgeAMA0GCSqGSIb3DQEBBAUAA4GBAMT6WtyFw95POY7UtF+YIYHiVRUf4SCq
hRIAGrljUePLo9iTqyPU1Pnt8JnIZ5P5BHU3MfgP8sqodaWub6mubkzaohJ1qD06
O87fnLCNid5Tov5jKogFHIki2EGGZxBosUw9lJlenQdNdDPbJc5LIWdfDvciA6jO
Nl8rOtKnt8Q+

---End - This line not part of the certificate request---

Redisplay enrollment request? [yes/no]:

Encryption key certificate request -

Certificate Request follows:

MIIBhTCB7wIBADAlMSMwIQYJKoZIhvcNAQkCFhRTYW5kQmFnZ2VyLmNpc2NvLmNv
bTCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAwG60QojpDbzbKnyj8FyTiOcv
THkDP7XD4vLT1XaJ409z0gSIoGnIcdFtXhVlBWtpq3/O9zYFXr1tH+BMCRQi3Lts
0IpxYa3D9iFPqev7SPXpsAIsY8a6FMq7TiwLObqiQjLKL4cbuV0Frjl0Yuv5A/Z+
kqMOm7c+pWNWFdLe9lsCAwEAAaAhMB8GCSqGSIb3DQEJDjESMBAwDgYDVR0PAQH/
BAQDAgUgMA0GCSqGSIb3DQEBBAUAA4GBACF7feURj/fJMojPBlR6fa9BrlMJx+2F
H91YM/CIiz2n4mHTeWTWKhLoT8wUfa9NGOk7yi+nF/F7035twLfq6n2bSCTW4aem
8jLMMaeFxwkrV/ceQKrucmNC1uVx+fBy9rhnKx8j60XE25tnp1U08r6om/pBQABU
eNPFhozcaQ/2

---End - This line not part of the certificate request---

Redisplay enrollment request? [yes/no]:

n

Router(config)#crypto pki import MS certificate

Enter the base 64 encoded certificate.

End with a blank line or the word "quit" on a line by itself

MIIDajCCAxSgAwIBAgIKFN7C6QAAAAAMRzANBgkqhkiG9w0BAQUFADA5MQswCQYD
VQQGEwJVUzEWMBQGA1UEChMNQ2lzY28gU3lzdGVtczESMBAGA1UEAxMJbXNjYS1y
b290MB4XDTAyMDYwODAxMTY0MloXDTAzMDYwODAxMjY0MlowJTEjMCEGCSqGSIb3
DQEJAhMUU2FuZEJhZ2dlci5jaXNjby5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0A
MIGJAoGBAMXYVxQ4lgJ/4SGbPc3zrbLCgHWqFmLtJrPRXvz3sNNXYdeL13cYGnLL
TrNj6+cJOoyzj8ab8TiT1skDOoqS5slcb6I4ci061aIa4eg56nL5Phv+aIbLPaPU
cbzjcMdONqUHIRZ8fRJDLmQu3r8EcSRKkZgR1wWfBpj942ELI0vDAgMBAAGjggHM
```

```
MIIByDALBgNVHQ8EBAMCB4AwHQYDVR0OBBYEFL8Quz8dyz4EGIeKx9A8UMNHLE4s
MHAGA1UdIwRpMGeAFKIacsl6dKAfuNDVQymlSp7esf8joT2kOzA5MQswCQYDVQQG
EwJVUzEWMBQGA1UEChMNQ2lzY28gU3lzdGVtczESMBAGA1UEAxMJbXNjYS1yb290
ghA6wKZelUfCh0qvJGipQtXuMCIGA1UdEQEB/wQYMBaCFFNhbmRCYWdnZXIuY2lz
Y28uY29tMG0GA1UdHwRmMGQwL6AtoCuGKWh0dHA6Ly9tc2NhLXJvb3QvQ2VydEVu
cm9sbC9tc2NhLXJvb3QuY3JsMDGgL6AthitmaWxlOi8vXFxtc2NhLXJvb3RcQ2Vy
dEVucm9sbFxtc2NhLXJvb3QuY3JsMIGUBggrBgEFBQcBAQSBhzCBhDA/BggrBgEF
BQcwAoYzaHR0cDovL21zY2Etcm9vdC9DZXJ0RW5yb2xsL21zY2Etcm9vdF9tc2Nh
LXJvb3QuY3J0MEEGCCsGAQUFBzAChjVmaWxlOi8vXFxtc2NhLXJvb3RcQ2VydEVu
cm9sbFxtc2NhLXJvb3RfbXNjYS1yb290LmNydDANBgkqhkiG9w0BAQUFAANBAJo2
r6sHPGBdTQX2EDoJpR/A2UHXxRYqVSHkFKZw0z31r5JzUM0oPNUETV7mnZlYNVRZ
CSEX/G8boi3WOjz9wZo=

% Router Certificate successfully imported

Router(config)#

Router(config)#crypto pki import MS certificate

Enter the base 64 encoded certificate.

End with a blank line or the word "quit" on a line by itself

MIIDajCCAxSgAwIBAgIKFN7OBQAAAAAMSDANBgkqhkiG9w0BAQUFADA5MQswCQYD
VQQGEwJVUzEWMBQGA1UEChMNQ2lzY28gU3lzdGVtczESMBAGA1UEAxMJbXNjYS1y
b290MB4XDTAyMDYwODAxMTY0NVoXDTAzMDYwODAxMjY0NVowJTEjMCEGCSqGSIb3
DQEJAhMUU2FuZEJhZ2dlci5jaXNjby5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0A
MIGJAoGBAMButEKI6Q282yp8o/Bck4jnL0x5Az+1w+Ly09V2ieNPc9IEiKBpyHHR
bV4VZQVraat/zvc2BV69bR/gTAkUIty7bNCKcWGtw/YhT6nr+0j16bACLGPGuhTK
u04sCzm6okIyyi+HG7ldBa45dGLr+QP2fpKjDpu3PqVjVhXS3vZbAgMBAAGjggHM
MIIByDALBgNVHQ8EBAMCBSAwHQYDVR0OBBYEFPDO29oRdlEUSgBMg6jZR+YFRWlj
MHAGA1UdIwRpMGeAFKIacsl6dKAfuNDVQymlSp7esf8joT2kOzA5MQswCQYDVQQG
EwJVUzEWMBQGA1UEChMNQ2lzY28gU3lzdGVtczESMBAGA1UEAxMJbXNjYS1yb290
ghA6wKZelUfCh0qvJGipQtXuMCIGA1UdEQEB/wQYMBaCFFNhbmRCYWdnZXIuY2lz
Y28uY29tMG0GA1UdHwRmMGQwL6AtoCuGKWh0dHA6Ly9tc2NhLXJvb3QvQ2VydEVu
cm9sbC9tc2NhLXJvb3QuY3JsMDGgL6AthitmaWxlOi8vXFxtc2NhLXJvb3RcQ2Vy
dEVucm9sbFxtc2NhLXJvb3QuY3JsMIGUBggrBgEFBQcBAQSBhzCBhDA/BggrBgEF
BQcwAoYzaHR0cDovL21zY2Etcm9vdC9DZXJ0RW5yb2xsL21zY2Etcm9vdF9tc2Nh
LXJvb3QuY3J0MEEGCCsGAQUFBzAChjVmaWxlOi8vXFxtc2NhLXJvb3RcQ2VydEVu
cm9sbFxtc2NhLXJvb3RfbXNjYS1yb290LmNydDANBgkqhkiG9w0BAQUFAANBAHaU
hyCwLirUghNxCmLzXRG7C3W1j0kSX7a4fX9OxKR/Z2SoMjdMNPPyApuh8SoT2zBP
ZKjZU2WjcZG/nZF4W5k=

% Router Certificate successfully imported
```

Certificate Autoenrollment Configuration Example

The following example shows how to configure the router to autoenroll with a CA on start-up:

```
Router(config)# crypto pki trustpoint frog
Router(ca-trustpoint)# enrollment url http://frog.phoobin.com/
Router(ca-trustpoint)# subject-name OU=Spiral Dept., O=tiedye.com
Router(ca-trustpoint)# ip-address ethernet-0
Router(ca-trustpoint)# auto-enroll regenerate
Router(ca-trustpoint)# password revokeme
Router(ca-trustpoint)# rsa-key frog 2048
!
Router(config)# crypto pki certificate chain frog
Router(config-cert-chain)# certificate ca 0B
30820293 3082023D A0030201 0202010B 300D0609 2A864886 F70D0101 04050030
79310B30 09060355 04061302 5553310B 30090603 55040813 02434131 15301306
0355040A 130C4369 73636F20 53797374 656D3120 301E0603 55040B13 17737562
6F726420 746F206B 6168756C 75692049 50495355 31243022 06035504 03131B79
6E692D75 31302043 65727469 66696361 7465204D 616E6167 6572301E 170D3030
30373134 32303536 32355A17 0D303130 37313430 31323834 335A3032 310E300C
06035504 0A130543 6973636F 3120301E 06092A86 4886F70D 01090216 11706B69
2D343562 2E636973 636F2E63 6F6D305C 300D0609 2A864886 F70D0101 01050003
4B003048 024100B3 0512A201 3B4243E1 378A9703 8AC5E3CE F77AF987 B5A422C4
15E947F6 70997393 70CF34D6 63A86B9C 4347A81A 0551FC02 ABA62360 01EF7DD2
6C136AEB 3C6C3902 03010001 A381F630 81F3300B 0603551D 0F040403 02052030
1C060355 1D110415 30138211 706B692D 3435622E 63697363 6F2E636F 6D301D06
03551D0E 04160414 247D9558 169B9A21 23D289CC 2DDA2A9A 4F77C616 301F0603
551D2304 18301680 14BD742C E892E819 1D551D91 683F6DB2 D8847A6C 73308185
0603551D 1F047E30 7C307AA0 3CA03AA4 38303631 0E300C06 0355040A 13054369
73636F31 24302206 03550403 131B796E 692D7531 30204365 72746966 69636174
65204D61 6E616765 72A23AA4 38303631 0E300C06 0355040A 13054369 73636F31
24302206 03550403 131B796E 692D7531 30204365 72746966 69636174 65204D61
6E616765 72300D06 092A8648 86F70D01 01040500 03410015 BC7CECF9 696697DF
E887007F 7A8DA24F 1ED5A785 C5C60452 47860061 0C18093D 08958A77 5737246B
0A25550A 25910E27 8B8B428E 32F8D948 3DD1784F 954C70
quit
```

Key Rollover for Certificate Renewal Configuration Examples

This section contains the following examples:

- Certificate Autoenrollment with Key Rollover Configuration Example, page 26-59
- Manual Certificate Enrollment with Key Rollover Configuration Example, page 26-60

Certificate Autoenrollment with Key Rollover Configuration Example

The following example shows how to configure the router to autoenroll with the CA named “trustme1” on startup. In this example, the regenerate keyword is issued, so a new key will be generated for the certificate. The renewal percentage is configured as 90 so if the certificate has a lifetime of one year, a new certificate is requested 36.5 days before the old certificate expires. The changes made to the running configuration are saved to the NVRAM startup configuration because autoenrollment will not update NVRAM if the running configuration has been modified but not written to NVRAM.

```
Router(config)# crypto pki trustpoint trustme1
Router(ca-trustpoint)# enrollment url http://trustme1.company.com/
Router(ca-trustpoint)# subject-name OU=Spiral Dept., O=tiedye.com
Router(ca-trustpoint)# ip-address ethernet0
Router(ca-trustpoint)# serial-number none
```

```
Router(ca-trustpoint)# auto-enroll 90 regenerate
Router(ca-trustpoint)# password revokeme
Router(ca-trustpoint)# rsakeypair trustme1 2048
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate trustme1
Router(config)# copy system:running-config nvram:startup-config
```

Manual Certificate Enrollment with Key Rollover Configuration Example

The following example shows how to configure key rollover to regenerate new keys with a manual certificate enrollment from the CA named “trustme2.”

```
Router(config)# crypto pki trustpoint trustme2
Router(ca-trustpoint)# enrollment url http://trustme2.company.com/
Router(ca-trustpoint)# subject-name OU=Spiral Dept., O=tiedye.com
Router(ca-trustpoint)# ip-address ethernet0
Router(ca-trustpoint)# serial-number none
Router(ca-trustpoint)# regenerate
Router(ca-trustpoint)# password revokeme
Router(ca-trustpoint)# rsakeypair trustme2 2048
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate trustme2
Router(config)# crypto pki enroll trustme2
Router(config)# exit
```

PKI: Query Multiple Servers During Certificate Revocation Check (CDP Override) Configuration Example

The following example uses the **match certificate override cdp** command to override the CDPs for the certificate map named Group1 defined in a **crypto pki certificate map** command:

```
Router(config)# crypto pki certificate map Group1 10
Router(ca-certificate-map)# subject-name co ou=WAN
Router(ca-certificate-map)# subject-name co o=Cisco
Router(config)# crypto pki trustpoint pki
Router(ca-trustpoint)# match certificate Group1 override cdp url http://server.cisco.com
```

Online Certificate Status Protocol Configuration Examples

This section provides the following configuration examples:

- OCSP Server Configuration Example, page 26-60
- CRL Then OCSP Server Configuration Example, page 26-61
- Specific OCSP Server Configuration Example, page 26-61

OCSP Server Configuration Example

The following example shows how to configure the router to use the OCSP server that is specified in the AIA extension of the certificate:

```
Router(config)# crypto pki trustpoint mytp
Router(ca-trustpoint)# revocation-check ocsp
```

CRL Then OCSP Server Configuration Example

The following example shows how to configure the router to download the CRL from the certificate distribution point (CDP); if the CRL is unavailable, the OCSP server that is specified in the AIA extension of the certificate will be used. If both options fail, certificate verification will also fail.

```
Router(config)# crypto pki trustpoint mytp
Router(ca-trustpoint)# revocation-check crl ocsp
```

Specific OCSP Server Configuration Example

The following example shows how to configure your router to use the OCSP server at the HTTP URL “http://myocspserver:81.” If the server is down, revocation check will be ignored.

```
Router(config)# crypto pki trustpoint mytp
Router(ca-trustpoint)# ocsp url http://myocspserver:81
Router(ca-trustpoint)# revocation-check ocsp none
```

Optional OCSP Nonces Configuration Example

The following example shows the unique identifier being disabled for OCSP communications for a previously created trustpoint named ts:

```
Router(config)# crypto pki trustpoint ts
Router (ca-truspoint)# ocsp disable-nonce
Router(ca-trustpoint)# end
```

Certificate Security Attribute-Based Access Control Configuration Example

The following example shows how to configure a certificate-based ACL:

```
Router(config)# crypto pki certificate map Group 10
Router(ca-certificate-map)# subject-name co Cisco
Router(config-cert-map)# exit
Router(config)# crypto pki trustpoint Access
Router(ca-trustpoint)# match certificate Group
Router(ca-trustpoint)# exit
set transform-set t2
match address 101
IRouter(config)# crypto keyring keyring1
Router(conf-keyring)# local-address serial2/0
Router(conf-keyring)# pre-shared-key address 10.0.0.1
Router(config)#
0:46:40: TYPE6_PASS: New Master key configured, encrypting the keys with
the new master key
Router(config)# exit
```

PKI AAA Authentication Using the Entire Subject Name Configuration Example

The following example shows that the entire subject name of the certificate is to be used for PKI AAA authentication.

```
Router(config)# aaa new-model
Router(config)# aaa authorization network tac-o group tacacs+

Router(config)# crypto pki trustpoint test
Router(ca-trustpoint)# enrollment url http://caserver:80
Router(ca-trustpoint)# revocation-check crl
Router(ca-trustpoint)# exit
Router(config)# authorization list tac-o
Router(config)# authorization username subjectname all

Router(config)# tacacs-server host 20.2.2.2 key a_secret_key
```

Source Interface Selection for Outgoing Traffic with Certificate Authority Configuration Example

In the following example, the router is located in a branch office. The router uses IP Security (IPSec) to communicate with the main office. Ethernet 1 is the "outside" interface that connects to the Internet Service Provider (ISP). Ethernet 0 is the interface connected to the LAN of the branch office. To access the CA server located in the main office, the router must send its IP datagrams out interface Ethernet 1 (address 10.2.2.205) using the IPSec tunnel. Address 10.2.2.205 is assigned by the ISP. Address 10.2.2.205 is not a part of the branch office or main office.

The CA cannot access any address outside the company because of a firewall. The CA sees a message coming from 10.2.2.205 and cannot respond (that is, the CA does not know that the router is located in a branch office at address 10.1.1.1, which it is able to reach).

Adding the **source interface** command tells the router to use address 10.1.1.1 as the source address of the IP datagram that it sends to the CA. The CA is able to respond to 10.1.1.1.

This scenario is configured using the **source interface** command and the interface addresses as described above.

```
Router(config)# crypto pki trustpoint ms-ca
Router(ca-trustpoint)# enrollment url http://ms-ca:80/certsrv/mscep/mscep.dll
Router(ca-trustpoint)# source interface ethernet0

Router(config)# interface ethernet 0
Router(config-if)# description inside interface
Router(config-if)# ip address 10.1.1.1 255.255.255.0

Router(config)# interface ethernet 1
Router(config-if)# description outside interface
Router(config-if)# ip address 10.2.2.205 255.255.255.0
Router(config-if)# crypto map main-office
```

Persistent Self-Signed Certificates Configuration Examples

The following examples show how to configure a persistent self-signed certificate:

- Trustpoint and Self-Signed Certificate Configuration Example, page 26-63
- Enabling the HTTPS Server Configuration Example, page 26-63

Trustpoint and Self-Signed Certificate Configuration Example

The following example shows how to configure a trustpoint and a self-signed certificate. In this example, a trustpoint named local is declared, its enrollment is requested, and a self-signed certificate with an IP address is generated:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# crypto pki trustpoint local
Router(ca-trustpoint)# enrollment selfsigned
Router(ca-trustpoint)# end
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# crypto pki enroll local
Nov 29 20:51:13.067: %SSH-5-ENABLED: SSH 1.99 has been enabled
Nov 29 20:51:13.267: %CRYPTO-6-AUTOGEN: Generated new 512 bit key pair
% Include the router serial number in the subject name? [yes/no]: yes
% Include an IP address in the subject name? [no]: yes
Enter Interface name or IP Address[]: ethernet 0
Generate Self Signed Router Certificate? [yes/no]: yes
Router Self Signed Certificate successfully created
```

Note A router can have only one self-signed certificate. If you attempt to enroll a trustpoint configured for a self-signed certificate and one already exists, you receive a notification and are asked if you want to replace it. If so, a new self-signed certificate is generated to replace the existing one.

Enabling the HTTPS Server Configuration Example

In the following example, the HTTPS server is enabled and a default trustpoint is generated because one was not previously configured:

```
Router(config)# ip http secure-server

% Generating 1024 bit RSA keys ...[OK]

*Dec 21 19:14:15.421:%PKI-4-NOAUTOSAVE:Configuration was modified. Issue "write memory"
to save new certificate

Router(config)#
```

Note You must save the configuration to NVRAM if you want to keep the self-signed certificate and have the HTTPS server enabled following router reloads.

The following message also appears:

```
*Dec 21 19:14:10.441:%SSH-5-ENABLED:SSH 1.99 has been enabled

Router(config)#
```

Note Creation of the key pair used with the self-signed certificate causes the Secure Shell (SSH) server to start. This behavior cannot be suppressed. You may want to modify your access control lists (ACLs) to permit or deny SSH access to the router.

Certificate Chain Verification Configuration Examples

The following examples show the possible output from the **crypto pki cert validate** command:

```
Router(config)# crypto pki cert validate ka

Validation Failed: trustpoint not found for ka

Router(config)# crypto pki cert validate ka

Validation Failed: can't get local certificate chain

Router(config)# crypto pki cert validate ka

Certificate chain has 2 certificates.
Certificate chain for ka is valid

Router(config)# crypto pki cert validate ka

Certificate chain has 2 certificates.
Validation Error: no certs on chain

Router(config)# crypto pki cert validate ka

Certificate chain has 2 certificates.
Validation Error: unspecified error
```

CHAPTER 27

Configuring Advanced VPNs Using the IPSec VPN SPA

This chapter provides information about configuring advanced IPSec VPNs on the IPSec VPN SPA on the Cisco 7600 series router. It includes the following sections:

Note The procedures in this chapter assume you have familiarity with security configuration concepts, such as VLANs, ISAKMP policies, preshared keys, transform sets, access control lists, and crypto maps. For more information about these and other security configuration concepts, refer to the *Cisco IOS Security Configuration Guide* and the *Cisco IOS Security Command Reference.*

For more information about the installation of cards on the Cisco 7600 series router, refer to the *Cisco 7600 Series Router SIP, SSC, and SPA Hardware Installation Guide* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/core/cis7600/76sipspa/sipspahw/index.htm

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Tip To ensure a successful configuration of your VPN using the IPSec VPN SPA, read all of the configuration summaries and guidelines before you perform any configuration tasks.

Overview of Advanced VPNs

Configuring IPSec VPNs in large, complicated networks can be quite complex. This chapter introduces features, such as Dynamic Multipoint VPN (DMVPN) and Easy VPN, which ease IPSec configuration in these advanced environments.

DMVPN allows users to better scale large and small IP Security (IPSec) Virtual Private Networks (VPNs) by combining generic routing encapsulation (GRE) tunnels, IPSec encryption, and Next Hop Resolution Protocol (NHRP).

Easy VPN allows a remote end user to communicate using IP Security (IPSec) with any Cisco IOS Virtual Private Network (VPN) gateway. Centrally managed IPSec policies are "pushed" to the client by the server, minimizing configuration by the end user.

Configuring DMVPN

The Dynamic Multipoint VPN (DMVPN) feature allows users to better scale large and small IP Security (IPSec) Virtual Private Networks (VPNs) by combining generic routing encapsulation (GRE) tunnels, IPSec encryption, and Next Hop Resolution Protocol (NHRP).

DMVPN Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring DMVPN:

- Only the Supervisor Engine 720 can be used for DMVPN hub or spoke configurations.
- A tunnel key should not be configured. If a tunnel key is configured, neither the PFC3 or the IPSec VPN SPA will take over the tunnel and the tunnel will be process switched.
- The hub configuration procedure varies based on which mode you are using: crypto-connect or VRF mode.
- GRE tunnels in different Virtual Routing and Forwarding (VRF) instances cannot share the same tunnel source.
- In crypto-connect mode, multipoint GRE tunnels should not share the same tunnel source.
- The following commands are not supported with multipoint GRE:
  - **ip tcp adjust-mss**
  - **qos pre-classify**
  - **tunnel vrf** *vrf_name*
  - **tunnel path-mtu-discovery**
- The Cisco 7600 series router cannot be a hub behind NAT.
- If the Cisco 7600 series router is a spoke, the hub cannot be behind NAT.
- If the Cisco 7600 series router is a hub, the spoke behind NAT must be either a Cisco 7600 series router or a device running software with caveat CSCef95695 resolved.
- If the Cisco 7600 series router is a spoke behind NAT, the hub must be either a Cisco 7600 series router or a device running software with caveat CSCef95695 resolved.
- Multicast streaming is not supported across DMVPN on a Cisco 7600 series router. Only multicast packets from a control plane such as routing protocols are supported.

- In a VRF-Aware DMVPN configuration, the **mls mpls tunnel-recir** command must be configured globally on the PE hub if the CE DMVPN spokes need to talk to other CEs across the MPLS cloud.
- In a VRF-Aware DMVPN configuration, a multipoint GRE interface should be configured with a large enough IP MTU (1416 packets) to prevent the router from doing fragmentation. This issue is tracked through caveat CSCeh37078.
- Before a multipoint GRE (mGRE) and IPSec tunnel can be established, you must define an Internet Key Exchange (IKE) policy by using the **crypto isakmp policy** command.
- For the NAT-Transparency Aware enhancement to work with DMVPN, you must use IPSec transport mode on the transform set. Also, even though NAT-Transparency (IKE and IPSec) can support two peers (IKE and IPSec) being translated to the same IP address (using the User Datagram Protocol [UDP] ports to differentiate them [this would be Peer Address Translation]), this functionality is not supported for DMVPN. All DMVPN spokes must have a unique IP address after they have been NAT translated. They can have the same IP address before they are NAT translated.
- If you use the Dynamic Creation for Spoke-to-Spoke Tunnels benefit of this feature, you must use IKE certificates or wildcard preshared keys for Internet Security Association and Key Management Protocol (ISAKMP) authentication.

Note It is highly recommended that you do not use wildcard preshared keys because access to the entire VPN is compromised if one spoke router is compromised.

- GRE tunnel keepalive (that is, the **keepalive** command under the GRE interface) is not supported on multipoint GRE tunnels
- Enhanced Interior Gateway Routing Protocol (EIGRP) should be avoided.
- Multiblade with DMVPN is not supported.
- DMVPN and IPSec stateful failover using HSRP and SSP cannot be configured simultaneously.
- DMVPN hierarchical hubs are not supported.

To enable multipoint GRE (mGRE) and IPSec tunneling for hub and spoke routers, you must configure a crypto map that uses a global IPSec policy template and configure your mGRE tunnel for IPSec encryption using the following procedures:

- Prerequisites, page 27-3
- Configuring an IPSec Profile, page 27-4
- Configuring the Hub for DMVPN in Crypto-Connect Mode, page 27-5
- Configuring the Hub for DMVPN in VRF Mode, page 27-6
- Configuring the Spoke for DMVPN, page 27-8
- Verifying the DMVPN Configuration, page 27-9

Prerequisites

Before configuring an IPSec profile, you must define a transform set by using the **crypto ipsec transform-set** command.

Before configuring the hub for DMVPN, you must choose a configuration mode: crypto-connect mode or VRF mode.

Configuring an IPSec Profile

The IPSec profile shares most of the same commands with the crypto map configuration, but only a subset of the commands are valid in an IPSec profile. Only commands that pertain to an IPSec policy can be issued under an IPSec profile; you cannot specify the IPSec peer address or the access control list (ACL) to match the packets that are to be encrypted.

To configure an IPSec profile, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto ipsec profile** *name*	Defines the IPSec parameters that are to be used for IPSec encryption between “spoke and hub” and “spoke and spoke” routers. This command enters crypto map configuration mode. • *name*—Name of the IPSec profile.
Step 2	Router(config-crypto-map)# **set transform-set** *transform-set-name*	Specifies which transform sets can be used with the IPSec profile. • *transform-set-name*—Name of the transform set.
Step 3	Router(config-crypto-map)# **set identity**	(Optional) Specifies identity restrictions to be used with the IPSec profile.
Step 4	Router(config-crypto-map)# **set security-association lifetime** {**seconds** *seconds* | **kilobytes** *kilobytes*}	(Optional) Overrides the global lifetime value for the IPSec profile. • *seconds*— Number of seconds a security association will live before expiring. • *kilobytes*— Volume of traffic (in kilobytes) that can pass between IPSec peers using a given security association before that security association expires.
Step 5	Router(config-crypto-map)# **set pfs** [**group1** | **group2**]	(Optional) Specifies that IP Security should ask for perfect forward secrecy (PFS) when requesting new security associations for this IPSec profile. If this command is not specified, the default (group1) will be enabled. • **group1**—(Optional) Specifies that IPSec should use the 768-bit Diffie-Hellman (DH) prime modulus group when performing the new DH exchange. • **group2**—(Optional) Specifies the 1024-bit DH prime modulus group.

Configuring the Hub for DMVPN in Crypto-Connect Mode

To configure the hub router for mGRE and IPSec integration (that is, associate the tunnel with the IPSec profile configured in the previous procedure) in crypto-connect mode, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface tunnel** *number*	Configures a tunnel interface and enters interface configuration mode • *number*—Number of the tunnel interface that you want to create or configure. There is no limit on the number of tunnel interfaces you can create.
Step 2	Router(config-if)# **ip address** *ip-address mask* [*secondary*]	Sets a primary or secondary IP address for the tunnel interface. • *ip-address*—IP address. • *mask*—Subnet mask. • [*secondary*]—(Optional) Secondary address.
Step 3	Router(config-if)# **ip mtu** *bytes*	Sets the maximum transmission unit (MTU) size, in bytes, of IP packets sent on an interface. • *bytes*—MTU size in bytes.
Step 4	Router(config-if)# **ip nhrp authentication** *string*	Configures the authentication string for an interface using the Next Hop Resolution Protocol (NHRP). • *string*—Text of the authentication string.
Step 5	Router(config-if)# **ip nhrp map multicast dynamic**	Allows NHRP to automatically add spoke routers to the multicast NHRP mappings.
Step 6	Router(config-if)# **ip nhrp network-id** *number*	Enables NHRP on an interface. • *number*—A globally unique 32-bit network identifier from a nonbroadcast multiaccess (NBMA) network. The range is from 1 to 4294967295.
Step 7	Router(config-if)# **tunnel source** {*ip-address* | *type number*}	Sets source address for a tunnel interface. • *ip-address*—IP address to use as the source address for packets in the tunnel. • *type number*—Interface type and number; for example, VLAN1.
Step 8	Router(config-if)# **tunnel mode gre multipoint**	Sets the encapsulation mode to mGRE for the tunnel interface.
Step 9	Router(config-if)# **tunnel protection ipsec profile** *name*	Associates a tunnel interface with an IPSec profile. • *name*—Name of the IPSec profile; this value must match the name specified in the **crypto ipsec profile** command.

	Command	Purpose
Step 10	Router(config-if)# **crypto engine slot** *slot/subslot*	Associates a tunnel interface tunnel protection socket with the IPSec VPN SPA. • *slot*/*subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 11	Router(config-if)# **interface vlan** *inside-vlan-id*	Enters interface configuration mode for the specified interface (or inside) VLAN interface. • *inside-vlan-id*—VLAN identifier that corresponds to an interface that has an IP address on the same network as the next hop leading to the tunnel destination.
Step 12	Router(config-if)# **crypto engine slot** *slot/subslot*	Associates the interface (or inside) VLAN with the IPSec VPN SPA . • *slot*/*subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 13	Router(config)# **interface vlan** *port-vlan-id*	Adds the specified VLAN interface and enters interface configuration mode for the specified port VLAN interface. • *port-vlan-id*—VLAN identifier that corresponds to the interface that encrypted packets will use to physically enter the router.
Step 14	Router(config-if)# **crypto connect vlan** *inside-vlan-id*	Connects the outside port VLAN to the inside interface VLAN and enters crypto-connect mode. • *inside-vlan-id*—Enter the VLAN identifier specified in Step 11.

Configuring the Hub for DMVPN in VRF Mode

To configure the hub router for mGRE and IPSec integration (that is, associate the tunnel with the IPSec profile configured in the previous procedure) in VRF mode, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface tunnel** *number*	Configures a tunnel interface and enters interface configuration mode • *number*—Number of the tunnel interface that you want to create or configure. There is no limit on the number of tunnel interfaces you can create.
Step 2	Router(config-if)# **ip address** *ip-address mask* [*secondary*]	Sets a primary or secondary IP address for the tunnel interface. • *ip-address*—IP address. • *mask*—Subnet mask. • [*secondary*]—(Optional) Secondary address.

	Command	Purpose
Step 3	Router(config-if)# **ip mtu** *bytes*	Sets the maximum transmission unit (MTU) size, in bytes, of IP packets sent on an interface. • *bytes*—MTU size in bytes.
Step 4	Router(config-if)# **ip nhrp authentication** *string*	Configures the authentication string for an interface using the Next Hop Resolution Protocol (NHRP). • *string*—Text of the authentication string.
Step 5	Router(config-if)# **ip nhrp map multicast dynamic**	Allows NHRP to automatically add spoke routers to the multicast NHRP mappings.
Step 6	Router(config-if)# **ip nhrp network-id** *number*	Enables NHRP on an interface. • *number*—A globally unique 32-bit network identifier from a nonbroadcast multiaccess (NBMA) network. The range is from 1 to 4294967295.
Step 7	Router(config-if)# **tunnel source** {*ip-address* | *type number*}	Sets source address for a tunnel interface. • *ip-address*—IP address to use as the source address for packets in the tunnel. • *type number*—Interface type and number; for example, VLAN1.
Step 8	Router(config-if)# **tunnel mode gre multipoint**	Sets the encapsulation mode to mGRE for the tunnel interface.
Step 9	Router(config-if)# **tunnel protection ipsec profile** *name*	Associates a tunnel interface with an IPSec profile. • *name*—Name of the IPSec profile; this value must match the name specified in the **crypto ipsec profile** command.
Step 10	Router(config-if)# **crypto engine slot** *slot/subslot*	Associates a tunnel interface tunnel protection socket with the IPSec VPN SPA. • *slot/subslot*—Enter the slot and subslot where the IPSec VPN SPA is located.
Step 11	Router(config-if)# **interface** *egress-interface*	Enters interface configuration mode for the specified interface. • *egress-interface*—Either a Layer 3 physical interface or the VLAN interface.
Step 12	Router(config-if)# **crypto engine slot** *slot/subslot*	Informs the egress interface that the incoming encrypted packets are to be processed by the IPSec VPN SPA. • *slot/subslot*—Enter the slot and subslot where the IPSec VPN SPA is located. This value must match the slot and subslot specified for the tunnel interface in Step 10.

Configuring the Spoke for DMVPN

To configure spoke routers for mGRE and IPSec integration, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **interface tunnel** *number*	Configures a tunnel interface and enters interface configuration mode • *number*—Number of the tunnel interface that you want to create or configure. There is no limit on the number of tunnel interfaces you can create.
Step 2	Router(config-if)# **ip address** *ip-address mask* [*secondary*]	Sets a primary or secondary IP address for the tunnel interface. • *ip-address*—IP address. • *mask*—Subnet mask. • [*secondary*]—(Optional) Secondary address.
Step 3	Router(config-if)# **ip mtu** *bytes*	Sets the maximum transmission unit (MTU) size, in bytes, of IP packets sent on an interface. • *bytes*—MTU size in bytes.
Step 4	Router(config-if)# **ip nhrp authentication** *string*	Configures the authentication string for an interface using NHRP. • *string*—Text of the authentication string.
Step 5	Router(config-if)# **ip nhrp map** *hub-tunnel-ip-address hub-physical-ip-address*	Statically configures the IP-to-NonBroadcast MultiAccess (NBMA) address mapping of IP destinations connected to an NBMA network. • *hub-tunnel-ip-address*—Defines the NHRP server at the hub, which is permanently mapped to the static public IP address of the hub. • *hub-physical-ip-address*—Defines the static public IP address of the hub.
Step 6	Router(config-if)# **ip nhrp map multicast** *hub-physical-ip-address*	Enables the use of a dynamic routing protocol between the spoke and hub, and sends multicast packets to the hub router. • *hub-physical-ip-address*—Defines the static public IP address of the hub.
Step 7	Router(config-if)# **ip nhrp nhs** *hub-tunnel-ip-address*	Configures the hub router as the NHRP next-hop server. • *hub-tunnel-ip-address*—Defines the NHRP server at the hub, which is permanently mapped to the static public IP address of the hub.

	Command	Purpose
Step 8	Router(config-if)# **ip nhrp network-id** *number*	Enables NHRP on an interface. • *number*—A globally unique 32-bit network identifier from a nonbroadcast multiaccess (NBMA) network. The range is from 1 to 4294967295.
Step 9	Router(config-if)# **tunnel source** {*ip-address* | *type number*}	Sets source address for a tunnel interface. • *ip-address*—IP address to use as the source address for packets in the tunnel. • *type number*—Interface type and number; for example, VLAN1.
Step 10	Router(config-if)# **tunnel mode gre multipoint**	Sets the encapsulation mode to mGRE for the tunnel interface. Use this command if data traffic can use dynamic spoke-to-spoke traffic.
Step 11	Router(config-if)# **tunnel protection ipsec profile** *name*	Associates a tunnel interface with an IPSec profile. • *name*—Name of the IPSec profile; this value must match the name specified in the **crypto ipsec profile** command.

Verifying the DMVPN Configuration

To verify that your Dynamic Multipoint VPN (DMVPN) configuration is working, enter the **show crypto isakmp sa addr**, **show crypto map**, and **show ip nhrp** commands.

The **show crypto isakmp sa addr** command displays all current IKE security associations (SAs) at a peer.

The following sample output is displayed after IKE negotiations have successfully completed between two peers.

```
Router# show crypto isakmp sa addr

dst             src             state          conn-id    slot

172.17.63.19    172.16.175.76   QM_IDLE              2       0

172.17.63.19    172.17.63.20    QM_IDLE              1       0

172.16.175.75   172.17.63.19    QM_IDLE              3       0
```

The **show crypto map** command displays the crypto map configuration.

The following sample output is displayed after a crypto map has been configured:

```
Router# show crypto map

Crypto Map "Tunnel5-head-0" 10 ipsec-isakmp

    Profile name: vpnprof
    Security association lifetime: 4608000 kilobytes/3600 seconds
    PFS (Y/N): N
    Transform sets={trans2, }

Crypto Map "Tunnel5-head-0" 20 ipsec-isakmp
```

```
    Map is a PROFILE INSTANCE.
    Peer = 172.16.175.75
    Extended IP access list
        access-list permit gre host 172.17.63.19 host 172.16.175.75
    Current peer: 172.16.175.75
    Security association lifetime: 4608000 kilobytes/3600 seconds
    PFS (Y/N): N
    Transform sets={trans2, }

Crypto Map "Tunnel5-head-0" 30 ipsec-isakmp

    Map is a PROFILE INSTANCE.
    Peer = 172.17.63.20
    Extended IP access list
        access-list permit gre host 172.17.63.19 host 172.17.63.20
    Current peer: 172.17.63.20
    Security association lifetime: 4608000 kilobytes/3600 seconds
    PFS (Y/N): N
    Transform sets={trans2, }

Crypto Map "Tunnel5-head-0" 40 ipsec-isakmp

    Map is a PROFILE INSTANCE.
    Peer = 172.16.175.76
    Extended IP access list
        access-list permit gre host 172.17.63.19 host 172.16.175.76
    Current peer: 172.16.175.76
    Security association lifetime: 4608000 kilobytes/3600 seconds
    PFS (Y/N): N
    Transform sets={trans2, }
    Interfaces using crypto map Tunnel5-head-0:

            Tunnel5
```

The **show ip nhrp** command displays the NHRP cache.

The following sample output shows that NHRP registration occurred, thereby allowing the user to apply tunnel protection:

```
Router# show ip nhrp

10.10.1.75/32 via 10.10.1.75, Tunnel5 created 00:32:11, expire 00:01:46

  Type: dynamic, Flags: authoritative unique registered

  NBMA address: 172.16.175.75

10.10.1.76/32 via 10.10.1.76, Tunnel5 created 00:26:41, expire 00:01:37

  Type: dynamic, Flags: authoritative unique registered

  NBMA address: 172.16.175.76

10.10.1.77/32 via 10.10.1.77, Tunnel5 created 00:31:26, expire 00:01:33

  Type: dynamic, Flags: authoritative unique registered

  NBMA address: 172.17.63.20
```

For complete configuration information for DMVPN support, refer to this URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080110ba1.html

For DMVPN configuration examples, see the "DMVPN Configuration Examples" section on page 27-13.

Configuring the Easy VPN Server

The Easy VPN Server provides server support for the Cisco VPN Client Release 4.x and later software clients and Cisco VPN hardware clients. The feature allows a remote end user to communicate using IP Security (IPSec) with any Cisco IOS Virtual Private Network (VPN) gateway. Centrally managed IPSec policies are "pushed" to the client by the server, minimizing configuration by the end user.

Easy VPN Server features include:

- Mode configuration and Xauth support
- User-based policy control
- Session monitoring for VPN group access
- RADIUS server support
- **backup-gateway** command
- **pfs** command
- Virtual IPsec interface support
- Banner, auto-update, and browser proxy
- Configuration management enhancements (pushing a configuration URL through a mode-configuration exchange)
- Per-user AAA policy download with PKI
- Syslog message enhancements
- Network admission control support

Easy VPN Server Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring the Easy VPN server:

- The following IPSec protocol options and attributes currently are not supported by Cisco VPN clients, so these options and attributes should not be configured on the router for these clients:
  - Authentication with public key encryption
  - Digital Signature Standard (DSS)
  - Diffie-Hellman (DH) groups (1)
  - IPSec Protocol Identifier (IPSEC_AH)
  - IPSec Protocol Mode (Transport mode)
  - Manual keys
  - Perfect Forward Secrecy (PFS)

For complete configuration information about the Easy VPN Server feature and the enhancements, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t8/ftunity.htm

Configuring the Easy VPN Remote

The Easy VPN feature allows a remote end user to communicate using IP Security (IPSec) with any Cisco IOS Virtual Private Network (VPN) gateway. Centrally managed IPSec policies are "pushed" to the client by the server, minimizing configuration by the end user.

Easy VPN Remote features include the following:

- Virtual IPsec interface support
- Banner, auto-update, and browser proxy
- Dual tunnel support
- Configuration management enhancements (pushing a configuration URL through a mode-configuration exchange)
- Reactivate primary peer

Easy VPN Remote Configuration Guidelines

Follow these guidelines when configuring Easy VPN for the IPSec VPN SPA:

Caution You need to clear all other crypto configurations from your running configuration on the Cisco IOS-based Easy VPN client that you are using to connect to the IPSec VPN SPA. If an ISAKMP policy is configured, it takes precedence over the pre-installed Easy VPN ISAKMP policies and the connection will fail. Other clients such as the VPN3000 and PIX systems running Easy VPN will prevent you from configuring Easy VPN unless all crypto configurations are removed. For complete configuration information for Easy VPN client support, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/ftezvpnr.htm

For an Easy VPN server configuration example, see the "Easy VPN Server (Router Side) Configuration Example" section on page 27-15.

Configuring Easy VPN Remote RSA Signature Storage

The Easy VPN Remote RSA Signature Support feature provides for the support of Rivest, Shamir, and Adelman (RSA) signatures on Easy VPN remote devices. The support is provided through RSA certificates that can be stored on or off the remote device.

Note The Easy VPN Remote RSA Signature Support feature is only supported as of Cisco IOS Release 12.2(33)SRA.

Easy VPN Remote RSA Signature Support Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Easy VPN Remote RSA Signature Support:

- You must have a Cisco Virtual Private Network (VPN) remote device and be familiar with configuring the device.
- You must have a certificate authority (CA) available to your network before you configure this interoperability feature. The CA must support the public key infrastructure (PKI) protocol of Cisco Systems, which is the Simple Certificate Enrollment Protocol (SCEP) (formerly called Certificate Enrollment Protocol [CEP]).
- This feature should be configured only when you also configure both IPSec and Internet Key Exchange (IKE) in your network.
- The Cisco IOS software does not support CA server public keys greater than 2048 bits.

Configuring Easy VPN Remote RSA Signature Support

The RSA signatures for an Easy VPN remote device are configured the same way that you would configure RSA signatures for any other Cisco device.

For information about configuring RSA signatures, refer to the *Cisco IOS Security Configuration Guide*.

To enable the RSA signatures, when you are configuring the Easy VPN remote and assigning the configuration to the outgoing interface, you must omit the **group** command. The content of the first Organizational Unit (OU) field will be used as the group.

For information about configuring Cisco Easy VPN remote devices, refer to the feature document, *Easy VPN Remote RSA Signature Support*, at the following location:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_7/gtevcrsa.htm

Configuration Examples

This section provides examples of the following configurations:

- DMVPN Configuration Examples, page 27-13
- Easy VPN Server (Router Side) Configuration Example, page 27-15

DMVPN Configuration Examples

The following sections provide examples of DMVPN configurations:

- IPSec VPN SPA Serving as a Hub in Crypto-Connect Mode Configuration Example, page 27-14
- IPSec VPN SPA Serving as a Hub in VRF Mode Configuration Example, page 27-14

IPSec VPN SPA Serving as a Hub in Crypto-Connect Mode Configuration Example

The following is a configuration example of the IPSec VPN SPA serving as a hub in crypto-connect mode:

```
interface Tunnel15
 ip address 10.10.100.15 255.255.0.0
 no ip redirects
 ip mtu 1416
!The following is required for EIGRP if a spoke-to-spoke tunnel is used.
 no ip next-hop-self eigrp 100
 ip nhrp authentication cisco
 ip nhrp map multicast dynamic
 ip nhrp network-id 100

!If ODR is used, you must manually enable CDP as it is disabled by default.
cdp enable
!Below is the interface VLAN's IP address.
tunnel source 172.16.15.2
tunnel mode gre multipoint
tunnel protection ipsec profile test
crypto engine slot 4/0

interface Vlan15
 ip address 172.16.15.2 255.255.255.0
 no mop enabled
 crypto engine slot 4/0

interface FastEthernet7/15
 no ip address
 crypto connect vlan 15
```

IPSec VPN SPA Serving as a Hub in VRF Mode Configuration Example

The following is a configuration example of the IPSec VPN SPA serving as a hub in VRF mode:

```
interface Tunnel110
 bandwidth 2000
 ip vrf forwarding red
 ip address 192.168.110.1 255.255.255.0
 no ip redirects
 ip mtu 1416
 ip nhrp authentication cisco
 ip nhrp map multicast dynamic
 ip nhrp network-id 110
 no ip split-horizon
 ip ospf network broadcast
 ip ospf priority 255
 tunnel source GigabitEthernet7/6
 tunnel mode gre multipoint
 tunnel protection ipsec profile test
 crypto engine slot 4/0

interface GigabitEthernet7/6
 ip address 172.16.6.2 255.255.255.0
 crypto engine slot 4/0

Router(config)# radius-server key nsite
```

Easy VPN Server (Router Side) Configuration Example

The following is an example of an Easy VPN server router-side configuration:

```
version 12.2
!hostname herckt
!
logging snmp-authfail
logging buffered 1000000 debugging
aaa new-model
aaa authentication login authen local
aaa authorization network author local
!
username unity password 0 uc
ip subnet-zero
no ip source-route
!
mpls ldp logging neighbor-changes
mls flow ip destination
mls flow ipx destination
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key 12345 address 0.0.0.0 0.0.0.0
crypto isakmp keepalive 10 2
!
crypto isakmp client configuration group group1
 key 12345
 domain cisco.com
 pool pool1
!
crypto isakmp client configuration group default
 key 12345
 domain cisco.com
 pool pool2
!
crypto ipsec transform-set myset3 esp-3des esp-md5-hmac
!
crypto dynamic-map test_dyn 1
 set transform-set myset3
 reverse-route
!
! Static client mapping
crypto map testtag client authentication list authen
crypto map testtag isakmp authorization list author
crypto map testtag client configuration address respond
crypto map testtag 10 ipsec-isakmp
 set peer 10.5.1.4
 set security-association lifetime seconds 900
 set transform-set myset3
 match address 109
!
! Dynamic client mapping
crypto map test_dyn client authentication list authen
crypto map test_dyn isakmp authorization list author
crypto map test_dyn client configuration address respond
crypto map test_dyn 1 ipsec-isakmp dynamic test_dyn
!
no spanning-tree vlan 513
!
redundancy
```

```
  main-cpu
  auto-sync running-config
  auto-sync standard
!
interface GigabitEthernet2/1
 no ip address
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,513,1002-1005
 switchport mode trunk
!
interface GigabitEthernet2/2
 no ip address
 shutdown
!
interface GigabitEthernet6/1
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,513,1002-1005
 switchport mode trunk
 cdp enable
!
interface GigabitEthernet6/2
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,2,1002-1005
 switchport mode trunk
 cdp enable
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan2
 no ip address
 crypto connect vlan 513
!
interface Vlan513
 ip address 10.5.1.1 255.255.0.0
 crypto map test_dyn
!
ip local pool pool1 22.0.0.2
ip local pool pool2 23.0.0.3
ip classless
ip pim bidir-enable
!
access-list 109 permit ip host 10.5.1.1 host 22.0.0.2
arp 127.0.0.12 0000.2100.0000 ARPA
!
snmp-server enable traps tty
snmp-server enable traps ipsec tunnel start
snmp-server enable traps ipsec tunnel stop
!
line con 0
line vty 0 4
 password lab
 transport input lat pad mop telnet rlogin udptn nasi
!
end
```

CHAPTER 28

Configuring Duplicate Hardware Configurations and IPSec Failover Using the IPSec VPN SPA

This chapter provides information about configuring duplicate hardware configurations and IPSec failover using the IPSec VPN SPA on the Cisco 7600 series router. It includes the following sections:

For detailed information on Cisco IOS IPSec cryptographic operations and policies, refer to the *Cisco IOS Security Configuration Guide* and *Cisco IOS Security Command Reference*.

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command References* and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Tip To ensure a successful configuration of your VPN using the IPSec VPN SPA, read all of the configuration summaries and guidelines before you perform any configuration tasks.

Overview of Duplicate Hardware Configurations and IPSec Failover

This chapter provides information about configuring duplicate hardware configurations and IPSec failover using the IPSec VPN SPA.

Duplicate Hardware Configurations

You can deploy up to ten IPSec VPN SPAs in a single chassis in crypto-connect mode or up to six IPSec VPN SPAs in a chassis in VRF mode, with the restriction that no more than one IPSec VPN SPA can be used to perform IPSec services for any given interface VLAN.

IPSec Failover

IPSec failover is a feature that increases the total uptime (or availability) of a customer's IPSec network. Traditionally, this is accomplished by employing a redundant (standby) router in addition to the original (active) router. If the active router becomes unavailable for any reason, the standby router takes over the processing of IKE and IPSec. IPSec failover falls into two categories: stateless failover and stateful failover.

IPSec stateless failover uses protocols such as the Hot Standby Router Protocol (HSRP) to provide primary to secondary cutover and also allows the active and standby VPN gateways to share a common virtual IP address. The drawback to this solution is that it requires the remote endpoints to detect that the gateway has gone down (using IKE keepalives or dead peer detection) and to completely re-establish IKE and IPSec sessions with the standby gateway. Although stateless failover meets some customers' needs, it does not accomplish transparent cutover to the backup device and typically results in lost application layer sessions due to the amount of time taken to re-establish sessions.

In contrast, IPSec stateful failover allows the active and standby routers to share IKE and IPSec state information so that each router has enough information to become the active router at any time. If the active router becomes unavailable for any reason, the standby router takes over the processing.

For Cisco IOS Release 12.2(33)SRA, the IPSec VPN SPA only supports IPSec stateful failover using a Blade Failure Group. A Blade Failure Group (BFG) allows two IPSec VPN SPAs to be installed in a chassis, with each IPSec VPN SPA serving as a backup for the other IPSec VPN SPA. A BFG is an active/active configuration. Both SPAs replicate data to each other so that either one can take over in the event of a failure.

Previous releases (that is, Cisco IOS Releases 12.2(18)SX) also support IPSec stateful failover using HSRP and State Synchronization Protocol (SSP) (also known as VPN High Availability). This feature enables a router to continue processing and forwarding packets after a planned or unplanned outage by employing a backup (standby) router that automatically takes over the primary (active) router's tasks in the event of an active router failure. The process is transparent to users and to remote IPSec peers. The time that it takes for the standby router to take over depends on HSRP timers.

Configuring Multiple IPSec VPN SPAs in a Chassis

You can deploy up to ten IPSec VPN SPAs in a single chassis in crypto-connect mode or up to six IPSec VPN SPAs in a chassis in VRF mode, with the restriction that no more than one IPSec VPN SPA can be used to perform IPSec services for any given interface VLAN.

Multiple IPSec VPN SPAs in a Chassis Configuration Guidelines

Follow these guidelines when configuring multiple IPSec VPN SPAs in a chassis:

- Note that using the **no switchport** command followed by the **switchport** command re-adds all VLANs to a trunk port (this situation occurs when you are first switching to a routed port and then back to a switch port). For detailed information on configuring trunk ports, see the "Configuring a Trunk Port" section on page 23-17.
- As with single IPSec VPN SPA deployments, you must properly configure each IPSec VPN SPA's inside and outside port. You can add an interface VLAN only to the inside port of one IPSec VPN SPA. Do not add the same interface VLAN to the inside port of more than one IPSec VPN SPA.

  Assigning interface VLANs to the inside ports of the IPSec VPN SPAs allows you to decide which IPSec VPN SPA can be used to provide IPSec services for a particular interface VLAN.

  Note It is not necessary to explicitly add interface VLANs to the inside trunk ports of the IPSec VPN SPAs. The **crypto engine slot** command achieves the same results.

  Note There is no support for using more than one IPSec VPN SPA to do IPSec processing for a single interface VLAN.

- SA-based load balancing is not supported.
- The **crypto map local address** command does not cause SA databases to be shared among multiple IPSec VPN SPAs.

For a multiple IPSec VPN SPAs in a chassis configuration example, see the "Multiple IPSec VPN SPAs in a Chassis Configuration Example" section on page 28-24.

Configuring IPSec Stateless Failover Using HSRP

The Hot Standby Routing Protocol (HSRP) is commonly used to provide failover between routers. HSRP tracks the state of router interfaces and provides a failover mechanism between primary and secondary devices. This functionality can be exploited to provide IPSec redundancy. HSRP has been coupled with RRI and IPSec to track state changes and provide a stateless IPSec failover mechanism. The Reverse Route Injection (RRI) feature is used to allow dynamic routing information updates during the HSRP and IPSec failover.

IPSec Stateless Failover Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring IPSec stateless failover

- Do not use IPSec stateless failover with tunnel protection in crypto-connect mode.

To configure IPSec stateless failover using HSRP, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp policy** *priority* **...** Router(config-isakmp) # **exit**	Defines an ISAKMP policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest. For details on configuring an ISAKMP policy, see the *Cisco IOS Security Configuration Guide*.
Step 2	Router(config)# **crypto isakmp key** *keystring* **address** *peer-address*	Configures a preshared authentication key. • *keystring*—Preshared key. • *peer-address*—IP address of the remote peer. For details on configuring a preshared key, see the *Cisco IOS Security Configuration Guide*.
Step 3	Router(config)# **crypto ipsec transform-set** *transform-set-name transform1*[*transform2*[*transform3*]] **...** Router(config-crypto-tran)# **exit**	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPSec security protocols and algorithms. For accepted *transformx* values, and more details on configuring transform sets, see the *Cisco IOS Security Command Reference*.

	Command	Purpose
Step 4	Router(config)# **access list** *access-list-number* {**deny** | **permit**} **ip** *source source-wildcard destination destination-wildcard*	Defines an extended IP access list. • *access-list-number*—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699. • {**deny** | **permit**}—Denies or permits access if the conditions are met. • *source*—Address of the host from which the packet is being sent. • *source-wildcard*—Wildcard bits to be applied to the source address. • *destination*—Address of the host to which the packet is being sent. • *destination-wildcard*—Wildcard bits to be applied to the destination address. For details on configuring an access list, see the *Cisco IOS Security Configuration Guide*.
Step 5	Router(config)# **crypto map** *map-name seq-number* **ipsec-isakmp** **...** Router(config-crypto-map)# **exit**	Creates or modifies a crypto map entry and enters the crypto map configuration mode. • *map-name*—Name that identifies the crypto map set. • *seq-number*—Sequence number you assign to the crypto map entry. Lower values have higher priority. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec security associations. For details on configuring a crypto map, see the *Cisco IOS Security Configuration Guide*.
Step 6	Router(config-if)# **interface gigabitethernet** *slot*/*subslot*/*port*	Enters interface configuration mode for the specified Gigabit Ethernet interface. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 7	Router(config-if)# **ip address** *ip-address mask*	Specifies the IP address and subnet mask for the interface. • *ip-address*—IP address. • *mask*—Subnet mask.

	Command	Purpose
Step 8	Router(config-if)# **standby** [*group-number*] **ip** *ip-address*	Enables the HSRP. • *group-number*—(Optional) Group number on the interface for which HSRP is being activated. The default is 0. The group number range is from 0 to 255 for HSRP version 1 and from 0 to 4095 for HSRP version 2. • *ip-address*—(Optional) IP address of the standby router interface.
Step 9	Router(config-if)# **standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime*	Configures the time between hello packets and the hold time before other routers declare the active router to be down. • *group-number*—(Optional) Group number to which the timers apply. • **msec**—(Optional) Interval in milliseconds. Millisecond timers allow for faster failover. • *hellotime*—Hello interval (in seconds). This is an integer from 1 to 254. The default is 3 seconds. If the **msec** option is specified, *hellotime* is in milliseconds. This is an integer from 15 to 999. • *holdtime*—Time (in seconds) before the active or standby router is declared to be down. This is an integer from x to 255. The default is 10 seconds. If the **msec** option is specified, *holdtime* is in milliseconds. This is an integer from y to 3000.

	Command	Purpose
Step 10	Router(config-if)# **standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** [**minimum** | **sync**] *seconds*]	Sets the standby priority used in choosing the active router. • *group-number*—(Optionbal) Group number to which the priority applies. • *priority*—(Optional) The priority value range is from 1 to 255, where 1 denotes the lowest priority and 255 denotes the highest priority. Specify that, if the local router has priority over the current active router, the local router should attempt to take its place as the active router. • **delay**—(Optional) Specifies a preemption delay, after which the Hot Standby router preempts and becomes the active router. • **minimum**—(Optional) Specifies the minimum delay period in seconds. • **sync**—(Optional) Specifies the maximum synchronization period for IP redundancy clients in seconds. • *seconds*—(Optional) Causes the local router to postpone taking over the active role for a minimum number of seconds since that router was last restarted. The range is from 0 to 3600 seconds (1 hour). The default is 0 seconds (no delay).
Step 11	Router(config-if)# **standby** [*group-number*] **track** *type number* [*interface-priority*]	Configures the interface to track other interfaces, so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered. • *group-number*—(Optional) Group number on the interface for which HSRP is being activated. • *type*—Interface type (combined with interface number) that will be tracked. • *number*—Interface number (combined with interface type) that will be tracked. • *interface-priority*—(Optional) Amount by which the Hot Standby priority for the router is decremented (or incremented) when the interface goes down (or comes back up). Range is from 0 to 255. Default is 10.
Step 12	Router(config-if)# **standby** [*group-number*] **name** *group-name*	Configures the standby group name for the interface. • *group-number*—(Optional) Group number to which the name is being applied. • **name** *group-name*—Name of the standby group.

	Command	Purpose
Step 13	Router(config-if)# **interface gigabitethernet** *slot*/*subslot*/*port*	Enters interface configuration mode for the specified Gigabit Ethernet interface. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 14	Router(config-if)# **ip address** *ip-address mask*	Specifies the IP address and subnet mask for the interface. • *ip-address*—IP address. • *mask*—Subnet mask.
Step 15	Router(config-if)# **standby** [*group-number*] **ip** *ip-address*	Enables the HSRP. • *group-number*—(Optional) Group number on the interface for which HSRP is being activated. The default is 0. The group number range is from 0 to 255 for HSRP version 1 and from 0 to 4095 for HSRP version 2. • *ip-address*—(Optional) IP address of the standby router interface.
Step 16	Router(config-if)# **standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime*	Configures the time between hello packets and the hold time before other routers declare the active router to be down. • *group-number*—(Optional) Group number to which the timers apply. • **msec**—(Optional) Interval in milliseconds. Millisecond timers allow for faster failover. • *hellotime*—Hello interval (in seconds). This is an integer from 1 to 254. The default is 3 seconds. If the **msec** option is specified, *hellotime* is in milliseconds. This is an integer from 15 to 999. • *holdtime*—Time (in seconds) before the active or standby router is declared to be down. This is an integer from x to 255. The default is 10 seconds. If the **msec** option is specified, *holdtime* is in milliseconds. This is an integer from y to 3000.

	Command	Purpose
Step 17	Router(config-if)# **standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** [**minimum** | **sync**] *seconds*]	Sets the standby priority used in choosing the active router. • *group-number*—(Optional) Group number to which the priority applies. • *priority*—(Optional) The priority value range is from 1 to 255, where 1 denotes the lowest priority and 255 denotes the highest priority. Specify that, if the local router has priority over the current active router, the local router should attempt to take its place as the active router. • **delay**—(Optional) Specifies a preemption delay, after which the Hot Standby router preempts and becomes the active router. • **minimum**—(Optional) Specifies the minimum delay period in seconds. • **sync**—(Optional) Specifies the maximum synchronization period for IP redundancy clients in seconds. • *seconds*—(Optional) Causes the local router to postpone taking over the active role for a minimum number of seconds since that router was last restarted. The range is from 0 to 3600 seconds (1 hour). The default is 0 seconds (no delay).
Step 18	Router(config-if)# **standby** [*group-number*] **track** *type number* [*interface-priority*]	Configures the interface to track other interfaces, so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered. • *group-number*—(Optional) Group number on the interface for which HSRP is being activated. • *type*—Interface type (combined with interface number) that will be tracked. • *number*—Interface number (combined with interface type) that will be tracked. • *interface-priority*—(Optional) Amount by which the Hot Standby priority for the router is decremented (or incremented) when the interface goes down (or comes back up). Range is from 0 to 255. Default is 10.

	Command	Purpose
Step 19	Router(config-if)# **standby** [*group-number*] **name** *group-name*	Configures the standby group name for the interface. • *group-number*—(Optional) Group number to which the name is being applied. • **name** *group-name*—Name of the standby group.
Step 20	Router(config-if)# **crypto map** *name* **redundancy** *standby-group-name*	Defines a backup IP Security (IPSec) peer. Both routers in the standby group are defined by the redundancy standby name and share the same virtual IP address. • *name*—Name that identifies the crypto map. • *standby-group-name*—Name of the standby group.

For examples of IPSec stateless failover configurations using HSRP, see "IPSec Stateless Failover Using HSRP Configuration Examples" section on page 28-26.

Configuring IPSec Stateless Failover in VRF Mode

Stateless failover is supported in VRF mode, but it is configured differently than in crypto-connect mode. In VRF mode, the HSRP configuration goes on the physical interface, but the crypto map is added to the interface VLAN. In crypto-connect mode, both the HSRP configuration and the crypto map are on the same interface.

For a configuration example of VRF mode stateless failover, see the "IPSec Stateless Failover in VRF Mode Configuration Example" section on page 28-31.

Configuring IPSec Stateful Failover Using HSRP and SSP

The IPSec Stateful Failover (VPN High Availability) feature enables a router to continue processing and forwarding packets after a planned or unplanned outage by employing a backup (standby) router that automatically takes over the primary (active) router's tasks in the event of an active router failure. The process is transparent to users and to remote IPSec peers. The time that it takes for the standby router to take over depends on HSRP timers.

Note IPSec Stateful Failover Using HSRP and SSP is *not* supported in Cisco IOS Release 12.2(33)SRA.

IPSec Stateful Failover (VPN High Availability) is designed to work in conjunction with the Hot Standby Router Protocol (HSRP), Reverse Route Injection (RRI) and the State Synchronization Protocol (SSP). When used together, HSRP, RRI, and SSP provide a more reliable network design for VPNs and reduce configuration complexity on remote peers.

IPSec Stateful Failover Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring IPSec stateful failover:

- When configuring IPSec stateful failover with the IPSec VPN SPA, note that all IPSec VPN SPA configuration rules apply. You must apply crypto maps to interface VLANs, and you must attach interface VLANs to the IPSec VPN SPA inside port.
- When configuring IPSec stateful failover with an IPSec VPN SPA in two chassis, note that the hardware configurations of both chassis must be exactly the same. For example, in one chassis if the IPSec VPN SPA that is in slot 2 is used to protect interface VLAN 100 and the IPSec VPN SPA that is in slot 3 is used to protect interface VLAN 101, the exact same configuration must be reflected in the second chassis. An example of a misconfiguration would be if the IPSec VPN SPA in slot 3 of the second chassis is used to protect interface VLAN 100.
- Do not use IPSec stateful failover with Easy VPN clients or IKE keepalives. IPSec stateful failover can be used with peers when DPD is used.
- Do not add nonexistent or inadequately configured HSRP standby groups to the State Synchronization Protocol (SSP) configuration because this action disables high-availability features until the configuration is corrected.
- Do not use the **standby use-bia** command. Always use a virtual HSRP MAC address for the router's MAC address.
- Do not use IPSec stateful failover with DMVPN or tunnel protection.
- The recommended HSRP timer values are one second for hello timers and three seconds for hold timers. These values should prevent an undesirable failover that is caused by temporary network congestion or transient, high CPU loads.

  These timer values can be adjusted upward if you are running high loads or have a large number of HSRP groups. Temporary failures and load-related system stability can be positively affected by raising the timer values as needed. The hello timer value should be approximately a third of the hold timer value.
- Use the HSRP "delay" timers to allow a device to finish booting, initializing, and synchronizing before participating as a high-availability pair. Set the "minimum" delay at 30 seconds or more to help prevent active/standby flapping and set the "reload" delay at some value greater than the minimum. You can use the delay timers to reflect the complexity and size of a particular configuration on various hardware. The delay timers tend to vary from platform to platform.
- Sequence number updates from active to standby have a 20-second minimum interval per SA.
- Due to dependence on HSRP, IPSec stateful failover does not work for secured WAN ports (IPSec over FlexWAN module port adapters).
- Use the Reverse Route Injection (RRI) feature **(reverse-route** command) to allow dynamic routing information updates during the HSRP and IPSec failover.
- After enabling both HSRP and IPSec stateful failover, use the **show ssp**, **show crypto ipsec**, and **show crypto isakmp** commands to verify that all processes are running properly.

To configure IP stateful failover using HSRP and SSP, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **ssp group** *group*	Indicates channel used to communicate High Availability (HA) information and enters SSP configuration mode. • *group*—Integer between 1 and 100.
Step 2	Router(config-ssp)# **redundancy** *name*	Identifies the HSRP group. • *name*—Valid IP redundancy group name.
Step 3	Router(config-ssp)# **remote** *ip-address*	Identifies peer that will receive High Availability (HA) transmissions. • *ip-address*—IP address of the standby router.
Step 4	Router(config)# **crypto isakmp policy** *priority* **...** Router(config-isakmp) # **exit**	Defines an ISAKMP policy and enters ISAKMP policy configuration mode. • *priority*—Identifies the IKE policy and assigns a priority to the policy. Use an integer from 1 to 10000, with 1 being the highest priority and 10000 the lowest. For details on configuring an ISAKMP policy, see the *Cisco IOS Security Configuration Guide*.
Step 5	Router(config)# **crypto isakmp key** *keystring* **address** *peer-address*	Configures a preshared authentication key. • *keystring*—Preshared key. • *peer-address*—IP address of the remote peer. For details on configuring a preshared key, see the *Cisco IOS Security Configuration Guide*.
Step 6	Router(config)# **crypto isakmp ssp** *id*	Enables ISAKMP state to be transferred by the SSP channel described by the ID. If this feature is disabled, all dormant SA entries bound to that ID on the standby router will be removed and any new state entries will not be added. • *id*—Channel used to transfer SA entries.
Step 7	Router(config)# **crypto ipsec transform-set** *transform-set-name transform1*[*transform2*[*transform3*]] **...** Router(config-crypto-tran)# **exit**	Defines a transform set (an acceptable combination of security protocols and algorithms) and enters crypto transform configuration mode. • *transform-set-name*—Name of the transform set. • *transform1*[*transform2*[*transform3*]]—Defines IPSec security protocols and algorithms. For accepted *transformx* values, and more details on configuring transform sets, see the *Cisco IOS Security Command Reference*.

	Command	Purpose
Step 8	Router(config)# **crypto map** *name* **ha replay-interval inbound** *inbound-interval* **outbound** *outbound-interval*	Specifies the intervals at which the active router should update the standby router with anti-replay sequence numbers. • *name*—Tag name of the crypto map described in the configuration. • *inbound-interval*—The interval at which the active router sends packet sequence updates for incoming packets. Integer between 0 and 10000. • *outbound-interval*—The interval at which the active router sends packet sequence updates for outgoing packets. Integer between 1 and 10 (in millions of packets).
Step 9	Router(config)# **access list** *access-list-number* {**deny** | **permit**} **ip** *source source-wildcard destination destination-wildcard*	Defines an extended IP access list. • *access-list-number*—Number of an access list. This is a decimal number from 100 to 199 or from 2000 to 2699. • {**deny** | **permit**}—Denies or permits access if the conditions are met. • *source*—Address of the host from which the packet is being sent. • *source-wildcard*—Wildcard bits to be applied to the source address. • *destination*—Address of the host to which the packet is being sent. • *destination-wildcard*—Wildcard bits to be applied to the destination address. For details on configuring an access list, see the *Cisco IOS Security Configuration Guide*.
Step 10	Router(config)# **crypto map** *map-name seq-number* **ipsec-isakmp** **...** Router(config-crypto-map)# **exit**	Creates or modifies a crypto map entry and enters the crypto map configuration mode. • *map-name*—Name that identifies the crypto map set. • *seq-number*—Sequence number you assign to the crypto map entry. Lower values have higher priority. • **ipsec-isakmp**—Indicates that IKE will be used to establish the IPSec security associations. For details on configuring a crypto map, see the *Cisco IOS Security Configuration Guide*.

	Command	Purpose
Step 11	Router(config-if)# **interface gigabitethernet** *slot/subslot/port*	Enters interface configuration mode for the specified Gigabit Ethernet interface. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 12	Router(config-if)# **ip address** *ip-address mask*	Specifies the IP address and subnet mask for the interface. • *ip-address*—IP address. • *mask*—Subnet mask.
Step 13	Router(config-if)# **standby** [*group-number*] **ip** *ip-address*	Enables the HSRP. • *group-number*—(Optional) Group number on the interface for which HSRP is being activated. The default is 0. The group number range is from 0 to 255 for HSRP version 1 and from 0 to 4095 for HSRP version 2. • *ip-address*—(Optional) IP address of the standby router interface.
Step 14	Router(config-if)# **standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime*	Configures the time between hello packets and the hold time before other routers declare the active router to be down. • *group-number*—(Optional) Group number to which the timers apply. • **msec**—(Optional) Interval in milliseconds. Millisecond timers allow for faster failover. • *hellotime*—Hello interval (in seconds). This is an integer from 1 to 254. The default is 3 seconds. If the **msec** option is specified, *hellotime* is in milliseconds. This is an integer from 15 to 999. • *holdtime*—Time (in seconds) before the active or standby router is declared to be down. This is an integer from x to 255. The default is 10 seconds. If the **msec** option is specified, *holdtime* is in milliseconds. This is an integer from y to 3000.

	Command	Purpose
Step 15	Router(config-if)# **standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** [**minimum** | **sync**] *seconds*]	Sets the standby priority used in choosing the active router. • *group-number*—(Optional) Group number to which the priority applies. • *priority*—(Optional) The priority value range is from 1 to 255, where 1 denotes the lowest priority and 255 denotes the highest priority. Specify that, if the local router has priority over the current active router, the local router should attempt to take its place as the active router. • **delay**—(Optional) Specifies a preemption delay, after which the Hot Standby router preempts and becomes the active router. • **minimum**—(Optional) Specifies the minimum delay period in seconds. • **sync**—(Optional) Specifies the maximum synchronization period for IP redundancy clients in seconds. • *seconds*—(Optional) Causes the local router to postpone taking over the active role for a minimum number of seconds since that router was last restarted. The range is from 0 to 3600 seconds (1 hour). The default is 0 seconds (no delay).
Step 16	Router(config-if)# **standby** [*group-number*] **track** *type number* [*interface-priority*]	Configures the interface to track other interfaces, so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered. • *group-number*—(Optional) Group number on the interface for which HSRP is being activated. • *type*—Interface type (combined with interface number) that will be tracked. • *number*—Interface number (combined with interface type) that will be tracked. • *interface-priority*—(Optional) Amount by which the Hot Standby priority for the router is decremented (or incremented) when the interface goes down (or comes back up). Range is from 0 to 255. Default is 10.
Step 17	Router(config-if)# **standby** [*group-number*] **name** *group-name*	Configures the standby group name for the interface. • *group-number*—(Optional) Group number to which the name is being applied. • **name** *group-name*—Name of the standby group.

	Command	Purpose
Step 18	Router(config-if)# **interface gigabitethernet** *slot*/*subslot*/*port*	Enters interface configuration mode for the specified Gigabit Ethernet interface. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA.
Step 19	Router(config-if)# **ip address** *ip-address mask*	Specifies the IP address and subnet mask for the interface. • *ip-address*—IP address. • *mask*—Subnet mask.
Step 20	Router(config-if)# **standby** [*group-number*] **ip** *ip-address*	Enables the HSRP. • *group-number*—(Optional) Group number on the interface for which HSRP is being activated. The default is 0. The group number range is from 0 to 255 for HSRP version 1 and from 0 to 4095 for HSRP version 2. • *ip-address*—(Optional) IP address of the standby router interface.
Step 21	Router(config-if)# **standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime*	Configures the time between hello packets and the hold time before other routers declare the active router to be down. • *group-number*—(Optional) Group number to which the timers apply. • **msec**—(Optional) Interval in milliseconds. Millisecond timers allow for faster failover. • *hellotime*—Hello interval (in seconds). This is an integer from 1 to 254. The default is 3 seconds. If the **msec** option is specified, *hellotime* is in milliseconds. This is an integer from 15 to 999. • *holdtime*—Time (in seconds) before the active or standby router is declared to be down. This is an integer from x to 255. The default is 10 seconds. If the **msec** option is specified, *holdtime* is in milliseconds. This is an integer from y to 3000.

	Command	Purpose
Step 22	Router(config-if)# **standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** [**minimum** | **sync**] *seconds*]	Sets the standby priority used in choosing the active router. • *group-number*—(Optional) Group number to which the priority applies. • *priority*—(Optional) The priority value range is from 1 to 255, where 1 denotes the lowest priority and 255 denotes the highest priority. Specify that, if the local router has priority over the current active router, the local router should attempt to take its place as the active router. • **delay**—(Optional) Specifies a preemption delay, after which the Hot Standby router preempts and becomes the active router. • **minimum**—(Optional) Specifies the minimum delay period in seconds. • **sync**—(Optional) Specifies the maximum synchronization period for IP redundancy clients in seconds. • *seconds*—(Optional) Causes the local router to postpone taking over the active role for a minimum number of seconds since that router was last restarted. The range is from 0 to 3600 seconds (1 hour). The default is 0 seconds (no delay).
Step 23	Router(config-if)# **standby** [*group-number*] **track** *type number* [*interface-priority*]	Configures the interface to track other interfaces, so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered. • *group-number*—(Optional) Group number on the interface for which HSRP is being activated. • *type*—Interface type (combined with interface number) that will be tracked. • *number*—Interface number (combined with interface type) that will be tracked. • *interface-priority*—(Optional) Amount by which the Hot Standby priority for the router is decremented (or incremented) when the interface goes down (or comes back up). Range is from 0 to 255. Default is 10.

	Command	Purpose
Step 24	Router(config-if)# **standby** [*group-number*] **name** *group-name*	Configures the standby group name for the interface. • *group-number*—(Optional) Group number to which the name is being applied. • **name** *group-name*—Name of the standby group.
Step 25	Router(config-if)# **crypto map name ssp** *id*	Enables IPSec state to be transferred by the SSP channel described by the ID. If this feature is disabled, all standby entries bound to that interface will be removed. • *id*—Channel used to transfer SA entries.

Verifying HSRP Configurations

To verify the IPSec stateful failover HSRP configuration, enter the **show crypto isakmp ha standby**, **show crypto ipsec ha**, **show crypto ipsec sa**, and **show crypto ipsec sa standby** commands.

Enter the **show crypto isakmp ha standby** command to view your ISAKMP standby or active SAs:

```
Router# show crypto isakmp ha standby

dst             src             state       I-Cookie           R-Cookie

172.16.31.100   20.3.113.1      QM_IDLE     796885F3 62C3295E  FFAFBACD EED41AFF

172.16.31.100   20.2.148.1      QM_IDLE     5B78D70F 3D80ED01  FFA03C6D 09FC50BE

172.16.31.100   20.4.124.1      QM_IDLE     B077D0A1 0C8EB3A0  FF5B152C D233A1E0

172.16.31.100   20.3.88.1       QM_IDLE     55A9F85E 48CC14DE  FF20F9AE DE37B913

172.16.31.100   20.1.95.1       QM_IDLE     3881DE75 3CF384AE  FF192CAB 795019AB
```

Enter the **show crypto ipsec ha** command to view your IPSec HA Manager state:

```
Router# show crypto ipsec ha

InterfaceVIPSAsIPSec Ha State

GigabitEthernet5/0/1172.16.31.1001800Active since 13:00:16 EDT Tue Oct 1 2002
```

Enter the **show crypto ipsec sa** command to view HA status of the IPSec SA (standby or active):

```
Router# show crypto ipsec sa

interface: GigabitEthernet5/0/1
   Crypto map tag: mymap, local addr. 172.168.3.100

    local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
    remote ident (addr/mask/prot/port): (5.6.0.0/255.255.0.0/0/0)
    current_peer: 172.168.3.1
    PERMIT, flags={}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0
```

```
    local crypto endpt.: 172.168.3.100, remote crypto endpt.: 172.168.3.1
    path mtu 1500, media mtu 1500
    current outbound spi: 132ED6AB

    inbound esp sas:
    spi: 0xD8C8635F(3637011295)
    transform: esp-des esp-md5-hmac ,
    in use settings ={Tunnel, }
    slot: 0, conn id: 2006, flow_id: 3, crypto map: mymap
    sa timing: remaining key lifetime (k/sec): (4499/59957)
    IV size: 8 bytes
    replay detection support: Y
    HA Status: STANDBY

  inbound ah sas:
    spi: 0xAAF10A60(2867923552)
    transform: ah-sha-hmac ,
    in use settings ={Tunnel, }
    slot: 0, conn id: 2004, flow_id: 3, crypto map: mymap
    sa timing: remaining key lifetime (k/sec): (4499/59957)
    replay detection support: Y
    HA Status: STANDBY

  inbound pcp sas:

  outbound esp sas:
    spi: 0x132ED6AB(321836715)
    transform: esp-des esp-md5-hmac ,
    in use settings ={Tunnel, }
    slot: 0, conn id: 2007, flow_id: 4, crypto map: mymap
    sa timing: remaining key lifetime (k/sec): (4499/59957)
    IV size: 8 bytes
    replay detection support: Y
    HA Status: STANDBY

  outbound ah sas:
    spi: 0x1951D78(26549624)
    transform: ah-sha-hmac ,
    in use settings ={Tunnel, }
    slot: 0, conn id: 2005, flow_id: 4, crypto map: mymap
    ssa timing: remaining key lifetime (k/sec): (4499/59957)
    replay detection support: Y
    HA Status: STANDBY

  outbound pcp sas:
```

Enter the **show crypto ipsec sa standby** command to view your standby SAs:

```
Router# show crypto ipsec sa standby

interface: GigabitEthernet5/0/1
    Crypto map tag: mymap, local addr. 172.168.3.100

    local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
    remote ident (addr/mask/prot/port): (5.6.0.0/255.255.0.0/0/0)
    current_peer: 172.168.3.1
    PERMIT, flags={}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0
```

```
local crypto endpt.: 172.168.3.100, remote crypto endpt.: 172.168.3.1
path mtu 1500, media mtu 1500
current outbound spi: 132ED6AB

inbound esp sas:
spi: 0xD8C8635F(3637011295)
transform: esp-des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 2006, flow_id: 3, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4499/59957)
IV size: 8 bytes
replay detection support: Y
HA Status: STANDBY

inbound ah sas:
spi: 0xAAF10A60(2867923552)
transform: ah-sha-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 2004, flow_id: 3, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4499/59957)
replay detection support: Y
HA Status: STANDBY

inbound pcp sas:

outbound esp sas:
spi: 0x132ED6AB(321836715)
transform: esp-des esp-md5-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 2007, flow_id: 4, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4499/59957)
IV size: 8 bytes
replay detection support: Y
HA Status: STANDBY

outbound ah sas:
spi: 0x1951D78(26549624)
transform: ah-sha-hmac ,
in use settings ={Tunnel, }
slot: 0, conn id: 2005, flow_id: 4, crypto map: mymap
sa timing: remaining key lifetime (k/sec): (4499/59957)
replay detection support: Y
HA Status: STANDBY

outbound pcp sas:
```

Displaying SSP Information

To verify the IPSec stateful failover SSP configuration, enter the **show ssp client**, **show ssp packet**, **show ssp peers**, and **show ssp redundancy** commands.

Enter the **show ssp client** command to view SSP client information:

```
Router# show ssp client

SSP Client Information

    DOI   Client Name                        Version   Running Ver

      1   IPSec HA Manager                    1.0       1.0

      2   IKE HA Manager                      1.0       1.0
```

Enter the **show ssp packet** command to view SSP packet information:

```
Router# show ssp packet

SSP packet Information

    Socket creation time: 01:01:06

    Local port: 3249      Server port: 3249

    Packets Sent = 38559, Bytes Sent = 2285020

    Packets Received = 910, Bytes Received = 61472
```

Enter the **show ssp peers** command to view SSP peer information:

```
Router# show ssp peers

SSP Peer Information

    IP Address       Connection State   Local Interface

    40.0.0.1         Connected          FastEthernet0/1
```

Enter the **show ssp redundancy** command to view redundancy information:

```
Router# show ssp redundancy

SSP Redundancy Information

  Device has been ACTIVE for 02:55:34

    Virtual IP       Redundancy Name              Interface

    172.16.31.100    KNIGHTSOFNI                  GigabitEthernet5/0/1GigabitEthernet0/0
```

For complete configuration information for Cisco IOS IPSec stateful failover support, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122limit/122y/122yx11/122_yx11/ipsecha.htm#wp1092482

For IPSec stateful failover configuration examples, see the "IPSec Stateful Failover Using HSRP and SSP Examples" section on page 28-35.

Configuring IPSec Stateful Failover Using a Blade Failure Group

This section describes how to configure IPSec stateful failover using a Blade Failure Group (BFG).

When two IPSec VPN SPAs are installed in a chassis, they are referred to as a Blade Failure Group (BFG). Each IPSec VPN SPA serves as a backup for the other IPSec VPN SPA. A BFG is an active/active configuration.

When an IPSec VPN SPA is joining a BFG or booting to come online, all of its IPSec and IKE data structures are synchronized with its peer. For each IPSec tunnel or IKE SA, and based on the per-interface crypto engine assignment, only one IPSec VPN SPA can be designated as active. For IKE SAs, an active SPA is the one that is accelerating cryptographic computations. For IPSec tunnels, the active SPA is the one that the traffic is passing through. For each IKE SA or IPSec tunnel, there is an active IPSec VPN SPA and its backup. For example, in a system that supports 1000 tunnels with two IPSec VPN SPAs, 500 of the tunnels may be active on one SPA and the remaining 500 may be active on the second SPA. Both SPAs then replicate data to each other so that either one can take over in the event of a failure. Each IPSec VPN SPA can have only one partner for all of the IKE and IPSec SAs that it protects.

IPSec Stateful Failover Using a BFG Configuration Guidelines

Follow these guidelines when configuring IPSec stateful failover using a BFG:

- Do not use IPSec stateful failover using a BFG in crypto-connect mode with tunnel protection.
- You can install or remove one of the IPSec VPN SPAs comprising a BFG without disrupting any of the tunnels on the other IPSec VPN SPA.

To configure IPSec stateful failover using a BFG, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **redundancy**	Enters redundancy configuration mode.
Step 2	Router(config-red)# **linecard-group** *group-number* **feature card**	Identifies the line card group ID for a Blade Failure Group and enters redundancy line card configuration mode. • *group-number*—Specifies a group ID for the BFG.
Step 3	Router(config-r-lc)# **subslot** *slot/subslot*	Adds the first SPA to the group. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed.
Step 4	Router(config-r-lc)# **subslot** *slot/subslot*	Adds the second SPA to the group. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed.

For an IPSec stateful failover using a BFG configuration example, see the "IPSec Stateful Failover Using a Blade Failure Group Configuration Example" section on page 28-39.

Verifying the IPSec Stateful Failover Using a BFG Configuration

To verify the IPSec stateful failover using a BFG configuration, enter the **show redundancy linecard-group** and **show crypto ace redundancy** commands.

Enter the **show redundancy linecard-group** command to display the components of a Blade Failure Group:

```
Router# show redundancy linecard-group 1

Line Card Redundancy Group:1 Mode:feature-card
Class:load-sharing
Cards:
Slot:3 Sublot:0
Slot:5 Sublot:0
```

Enter the **show crypto ace redundancy** command to display information about a Blade Failure Group:

```
Router# show crypto ace redundancy

--------------------------------------
LC Redundancy Group ID            :1
Pending Configuration Transactions:0
Current State                     :OPERATIONAL
Number of blades in the group     :2
Slots
--------------------------------------
Slot:3 subslot:0
Slot state:0x36
Booted
Received partner config
Completed Bulk Synchronization
Crypto Engine in Service
Rebooted 22 times
Initialization Timer not running
Slot:5 subslot:0
Slot state:0x36
Booted
Received partner config
Completed Bulk Synchronization
Crypto Engine in Service
Rebooted 24 times
Initialization Timer not running

ACE B2B Group State:OPERATIONAL Event:BULK DONE
ACE B2B Group State:CREATED Event:CONFIG_DOWNLOAD_DONE
ACE B2B Group State:DELETED Event:CONFIG_DELETE
ACE B2B Group State:OPERATIONAL Event:BULK DONE
ACE B2B Group State:CREATED Event:CONFIG_DOWNLOAD_DONE
ACE B2B Group State:DELETED Event:CONFIG_DELETE
ACE B2B Group State:OPERATIONAL Event:CONFIG_DOWNLOAD_DONE
ACE B2B Group State:DELETED Event:CONFIG_ADD
ACE B2B Group State:CREATED Event:UNDEFINED B2B HA EVENT
ACE B2B Group State:CREATED Event:CONFIG_DOWNLOAD_DONE
```

Configuration Examples

This section provides examples of the following configurations:

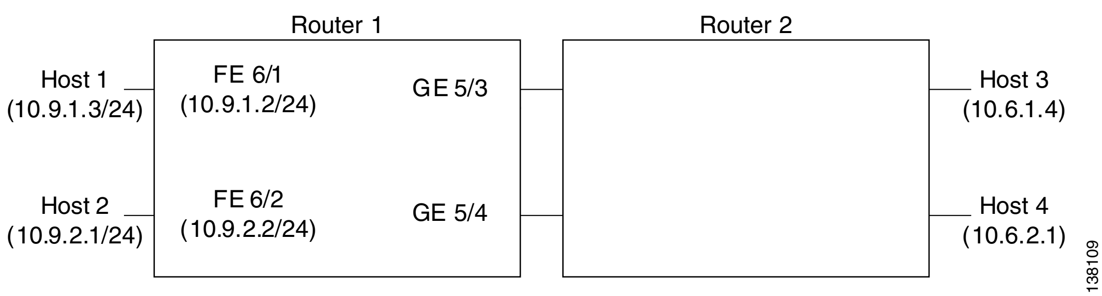
- Multiple IPSec VPN SPAs in a Chassis Configuration Example, page 28-24
- IPSec Stateless Failover Using HSRP Configuration Examples, page 28-26
- IPSec Stateless Failover in VRF Mode Configuration Example, page 28-31
- IPSec Stateful Failover Using HSRP and SSP Examples, page 28-35
- IPSec Stateful Failover Using a Blade Failure Group Configuration Example, page 28-39

Multiple IPSec VPN SPAs in a Chassis Configuration Example

This section provides an example of a configuration using multiple IPSec VPN SPAs in a chassis as shown in Figure 28-1. Note the following in these examples:

- An IPSec VPN SPA is in slot 2, subslot 0 and slot 3, subslot 0 of router 1.
- In the configuration example, three exclamation points (!!!) precede descriptive comments.

Figure 28-1 Multiple IPSec VPN SPAs in a Chassis Configuration Example




```
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key mykey address 10.8.1.1
crypto isakmp key mykey address 10.13.1.1
!
crypto ipsec transform-set xform1 ah-md5-hmac esp-des esp-sha-hmac
crypto ipsec transform-set xform2 esp-3des esp-sha-hmac
!
!!! crypto map applied to VLAN 12, which is
!!! assigned to "inside" port of IPSec VPN SPA in slot 3
crypto map cmap2 10 ipsec-isakmp
 set peer 10.8.1.1
 set transform-set xform1
 match address 102
!
!!! crypto map applied to VLAN 20, which is
!!! assigned to "inside" port of IPSec VPN SPA in slot 2/0
crypto map cmap3 10 ipsec-isakmp
 set peer 10.13.1.1
 set transform-set xform2
```

```
 match address 103
!
!!! "port" VLAN, crypto connected to VLAN 12 by IPSec VPN SPA on slot 3/0
interface Vlan11
 no ip address
 crypto connect vlan 12
!
!!! "interface" VLAN, assigned to IPSec VPN SPA on slot 3/0
interface Vlan12
 ip address 10.8.1.2 255.255.0.0
 crypto map cmap2
 crypto engine slot 3/0
!
!!! "port" VLAN, crypto connected to VLAN 20 by IPSec VPN SPA on slot 2/0
interface Vlan19
 no ip address
 crypto connect vlan 20
!
!!! "interface" VLAN, assigned to IPSec VPN SPA on slot 2/0
interface Vlan20
 ip address 10.13.1.2 255.255.0.0
 crypto map cmap3
 crypto engine slot 2/0
!
!!! connected to Host 1
interface FastEthernet6/1
 ip address 10.9.1.2 255.255.255.0
!
!!! connected to Host 2
interface FastEthernet6/2
 ip address 10.9.2.2 255.255.255.0
!
!!! connected to Router 2
interface GigabitEthernet5/3
 switchport
 switchport mode access
 switchport access vlan 11
!
!!! connected to Router 2
interface GigabitEthernet5/4
 switchport
 switchport mode access
 switchport access vlan 19
!
interface GigabitEthernet2/0/1
 no ip address
 flowcontrol receive on
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 12,1002-1005
 switchport mode trunk
 cdp enable
!
interface GigabitEthernet2/0/2
 no ip address
 flowcontrol receive on
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 11,1002-1005
 switchport mode trunk
 cdp enable
!
interface GigabitEthernet3/0/1
 no ip address
```

```
 flowcontrol receive on
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,1002-1005
 switchport mode trunk
 cdp enable
!
interface GigabitEthernet3/0/2
 no ip address
 flowcontrol receive on
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 19,1002-1005
 switchport mode trunk
 cdp enable
!
ip classless
!
!!! packets from Host 1 to Host 3 are routed from FastEthernet6/1
!!! to VLAN 12, encrypted with crypto map cmap2
!!! using IPSec VPN SPA in slot 3/0, and forwarded to peer 10.8.1.1
!!! through GigabitEthernet5/3
ip route 10.6.1.4 255.255.255.255 10.8.1.1
!
!!! packets from Host 2 to Host 4 are routed from FastEthernet6/2
!!! to VLAN 20, encrypted with crypto map cmap3
!!! using IPSec VPN SPA in slot 2/0, and forwarded to peer 10.13.1.1
!!! through GigabitEthernet5/4
ip route 10.6.2.1 255.255.255.255 10.13.1.1
!
!!! ACL matching traffic between Host 1 and Host 3
access-list 102 permit ip host 10.9.1.3 host 10.6.1.4
!
!!! ACL matching traffic between Host 2 and Host 4
access-list 103 permit ip host 10.9.2.1 host 10.6.2.1
```

IPSec Stateless Failover Using HSRP Configuration Examples

This section provides the following configuration examples of IPSec stateless failover using HSRP:

- IPSec Stateless Failover Using HSRP for the Active Chassis Configuration Example, page 28-26
- IPSec Stateless Failover Using HSRP for the Standby Chassis Configuration Example, page 28-28
- IPSec Stateless Failover Using HSRP for the Remote Router Configuration Example, page 28-30

IPSec Stateless Failover Using HSRP for the Active Chassis Configuration Example

The following example shows the configuration for an active chassis that is configured for IPSec stateless failover using HSRP:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Active
!
redundancy
 main-cpu
  auto-sync standard
```

```
ip subnet-zero
!
no ip domain-lookup
!
no mls ip multicast aggregate
no mls ip multicast non-rpf cef
!
crypto isakmp policy 1
 encr 3des
 authentication pre-share
crypto isakmp key NEEWOMM address 0.0.0.0 0.0.0.0
!
crypto ipsec security-association lifetime seconds 86400
!
crypto ipsec transform-set TS1 esp-3des esp-sha-hmac
!
crypto map ha 10 ipsec-isakmp
 set peer 172.16.31.3
 set transform-set TS1
 match address 101
!
spanning-tree extend system-id
no spanning-tree vlan 4
!
interface GigabitEthernet1/1
 no ip address
 no ip redirects
 crypto connect vlan 4
!
interface GigabitEthernet1/2
 ip address 40.0.0.1 255.255.255.0
 no ip redirects
 standby delay minimum 35 reload 60
 standby ip 40.0.0.100
 standby timers 3 5
 standby preempt
 standby track GigabitEthernet1/1
  standby track vlan 4
!
interface GigabitEthernet5/1/1
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,4,1002-1005
 switchport mode trunk
 flowcontrol receive on
 cdp enable
!
interface GigabitEthernet5/1/2
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 flowcontrol receive on
 cdp enable
!
interface Vlan1
 no ip address
 shutdown
```

```
!
interface Vlan4
 ip address 172.16.31.1 255.255.255.0
 standby delay minimum 35 reload 60
 standby ip 172.16.31.100
 standby timers 3 5
 standby preempt
 standby name KNIGHTSOFNI
 standby track GigabitEthernet1/1
 standby track GigabitEthernet1/2
 crypto map ha redundancy KNIGHTSOFNI
 crypto engine slot 5/1
!
ip classless
ip route 10.11.1.1 255.255.255.255 172.16.31.3
no ip http server
ip pim bidir-enable
!
access-list 101 permit ip host 40.0.0.3 host 10.11.1.1
arp 127.0.0.12 0000.2100.0000 ARPA
!
line con 0
line vty 0 4
 login
 transport input lat pad mop telnet rlogin udptn nasi ssh
!
end
```

IPSec Stateless Failover Using HSRP for the Standby Chassis Configuration Example

The following example shows the configuration for a standby chassis that is configured for IPSec stateless failover using HSRP:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname StandBy
!
redundancy
 main-cpu
  auto-sync standard
ip subnet-zero
!
no ip domain-lookup
!
no mls ip multicast aggregate
no mls ip multicast non-rpf cef
!
crypto isakmp policy 1
 encr 3des
 authentication pre-share
crypto isakmp key NEEWOMM address 0.0.0.0 0.0.0.0
!
crypto ipsec security-association lifetime seconds 86400
!
crypto ipsec transform-set TS1 esp-3des esp-sha-hmac
!
crypto map ha 10 ipsec-isakmp
 set peer 172.16.31.3
 set transform-set TS1
```

```
 match address 101
!
spanning-tree extend system-id
no spanning-tree vlan 4
!
interface GigabitEthernet1/1
 no ip address
 no ip redirects
 crypto connect vlan 4
!
interface GigabitEthernet1/2
 ip address 40.0.0.2 255.255.255.0
 no ip redirects
 standby delay minimum 35 reload 60
 standby ip 40.0.0.100
 standby timers 3 5
 standby preempt
 standby track GigabitEthernet1/1
  standby track vlan 4
!
interface GigabitEthernet5/1/1
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,4,1002-1005
 switchport mode trunk
 flowcontrol receive on
 cdp enable
!
interface GigabitEthernet5/1/2
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 flowcontrol receive on
 cdp enable
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan4
 ip address 172.16.31.2 255.255.255.0
 standby delay minimum 35 reload 60
 standby ip 172.16.31.100
 standby timers 1 3
 standby preempt
 standby name KNIGHTSOFNI
 standby track GigabitEthernet1/1
 standby track GigabitEthernet1/2
 crypto map ha redundancy KNIGHTSOFNI
  crypto engine slot 5/1
!
ip classless
ip route 10.11.1.1 255.255.255.255 172.16.31.3
no ip http server
ip pim bidir-enable
!
access-list 101 permit ip host 40.0.0.3 host 10.11.1.1
```

```
arp 127.0.0.12 0000.2100.0000 ARPA
!
line con 0
line vty 0 4
 login
 transport input lat pad mop telnet rlogin udptn nasi ssh
!
end
```

IPSec Stateless Failover Using HSRP for the Remote Router Configuration Example

The following example shows the configuration for a remote router that is configured for IPSec stateless failover using HSRP. Note that the router in this example is not using an IPSec VPN SPA.

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RemotePeer
!
redundancy
 main-cpu
  auto-sync standard
ip subnet-zero
!
no ip domain-lookup
!
no mls ip multicast aggregate
no mls ip multicast non-rpf cef
!
crypto isakmp policy 1
 encr 3des
 authentication pre-share
crypto isakmp key NEEWOMM address 0.0.0.0 0.0.0.0
!
crypto ipsec security-association lifetime seconds 86400
!
crypto ipsec transform-set TS1 esp-3des esp-sha-hmac
!
crypto map ha 10 ipsec-isakmp
 set peer 172.16.31.100
 set transform-set TS1
 match address 101
!
spanning-tree extend system-id
!
interface Loopback1
 ip address 10.11.1.1 255.255.255.0
!
interface GigabitEthernet1/1
 no ip address
 shutdown
!
interface GigabitEthernet1/2
 ip address 172.16.31.3 255.255.0.0
 crypto map ha
!
interface GigabitEthernet3/1
 mtu 4500
 no ip address
  flowcontrol receive on
```

```
 cdp enable
!
interface GigabitEthernet3/2
 mtu 4500
 no ip address
  flowcontrol receive on
 cdp enable
!
interface Vlan1
 no ip address
 shutdown
!
ip classless
ip route 40.0.0.3 255.255.255.255 172.16.31.100
no ip http server
ip pim bidir-enable
!
access-list 101 permit ip host 10.11.1.1 host 40.0.0.3
arp 127.0.0.12 0000.2100.0000 ARPA
!
line con 0
line vty 0 4
 login
 transport input lat pad mop telnet rlogin udptn nasi ssh
!
end
```

IPSec Stateless Failover in VRF Mode Configuration Example

The following example shows a VRF mode configuration with chassis-to-chassis IPSec stateless failover:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service internal
service counters max age 10
!
hostname router-Charlie
!
logging snmp-authfail
logging buffered 100000 debugging
enable password lab
!
no aaa new-model
ip subnet-zero
!
no ip domain-lookup
ip host tftp 223.255.254.254
!
ip vrf coke
 rd 2000:1
 route-target export 2000:1
 route-target import 2000:1
!
ip vrf pepsi
 rd 1000:1
 route-target export 1000:1
 route-target import 1000:1
!
mls ip multicast flow-stat-timer 9
```

```
no mls flow ip
no mls flow ipv6
mls cef error action freeze
!
crypto keyring key0
  pre-shared-key address 0.0.0.0 0.0.0.0 key NEEWOMM
!
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
 lifetime 7200
!
crypto isakmp profile prof1
   vrf coke
   keyring key0
   match identity address 1.1.1.2 255.255.255.255
crypto isakmp profile prof2
   vrf pepsi
   keyring key0
   match identity address 1.1.1.2 255.255.255.255
!
crypto ipsec transform-set TR esp-3des esp-md5-hmac
!
crypto ipsec profile tunpro
 set transform-set TR
 set isakmp-profile prof1
!
crypto map M10k local-address FastEthernet3/39
crypto map M10k 1 ipsec-isakmp
 set peer 1.1.1.2
 set transform-set TR
 set isakmp-profile prof1
 match address 110
!
crypto map M10k2 local-address FastEthernet3/39
crypto map M10k2 1 ipsec-isakmp
 set peer 1.1.1.2
 set transform-set TR
 set isakmp-profile prof2
 match address 111
!
crypto engine mode vrf
!
no power enable module 4
!
power redundancy-mode combined
spanning-tree mode pvst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
diagnostic cns publish cisco.cns.device.diag_results
diagnostic cns subscribe cisco.cns.device.diag_commands
!
redundancy
 mode rpr-plus
 linecard-group 1 feature-card
  class load-sharing
 main-cpu
  auto-sync running-config
  auto-sync standard
!
vlan internal allocation policy ascending
vlan access-log ratelimit 2000
```

```
!
interface Tunnel39
 ip address 50.0.0.2 255.0.0.0
 shutdown
 tunnel source FastEthernet3/39
 tunnel destination 192.39.1.1
!
interface FastEthernet3/1
 description connected to pocono-lnx eth3
 no ip address
 switchport
 switchport access vlan 102
 switchport mode access
!
interface FastEthernet3/2
 description connected to pocono-lnx eth2
 no ip address
 shutdown
!
interface FastEthernet3/3
 no ip address
 shutdown
.
.
.
!
interface FastEthernet3/36
 no ip address
 shutdown
!
interface FastEthernet3/37
 no ip address
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,102,202,1002-1005
 switchport mode trunk
!
interface FastEthernet3/38
 no ip address
 shutdown
!
interface FastEthernet3/39
 ip address 1.1.1.12 255.255.255.0
 standby delay minimum 30 reload 90
 standby 1 ip 1.1.1.1
 standby 1 timers 1 4
 standby 1 preempt
 standby 1 name PUBLIC
 standby 1 track Vlan100
 standby 1 track Vlan102
 crypto engine slot 5/0
!
interface FastEthernet3/40
 no ip address
 shutdown
.
.
.
!
interface FastEthernet3/47
 no ip address
 shutdown
!
interface FastEthernet3/48
```

```
 ip address 17.16.16.2 255.255.0.0
!
interface GigabitEthernet4/1
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet4/2
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet5/0/1
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,100,200,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet5/0/2
 no ip address
 flowcontrol receive on
 flowcontrol send off
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 spanning-tree portfast trunk
!
interface GigabitEthernet6/1
 no ip address
 shutdown
!
interface GigabitEthernet6/2
 no ip address
 shutdown
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan100
 ip vrf forwarding coke
 ip address 3.3.3.2 255.255.255.0
 crypto map M10k red PUBLIC
 crypto engine slot 5/0
!
interface Vlan102
 ip vrf forwarding coke
 ip address 10.83.3.5 255.255.255.0
 standby delay minimum 30 reload 90
```

```
 standby 2 ip 10.83.3.2
 standby 2 preempt
 standby 2 name VPNSM
 standby 2 track Vlan100
 standby 2 track Vlan102
!
interface Vlan200
 ip vrf forwarding pepsi
 ip address 3.3.3.2 255.255.255.0
 crypto map M10k2 red PUBLIC
 crypto engine slot 5/0
!
interface Vlan202
 ip vrf forwarding pepsi
 ip address 10.83.3.5 255.255.255.0
 standby delay minimum 30 reload 90
 standby 3 ip 10.83.3.2
 standby 3 preempt
 standby 3 name VPNSM1
 standby 3 track Vlan200
 standby 3 track Vlan202
!
ip classless
ip route 223.255.254.253 255.255.255.255 17.16.0.1
ip route 223.255.254.254 255.255.255.255 17.16.0.1
ip route vrf coke 4.4.4.0 255.255.255.0 Vlan100
ip route vrf coke 10.10.20.0 255.255.255.0 Vlan100
ip route vrf pepsi 4.4.4.0 255.255.255.0 Vlan200
no ip http server
!
access-list 110 permit ip any host 4.4.4.2
access-list 111 permit ip any host 4.4.4.3
access-list 120 permit ip host 10.83.3.1 host 10.10.20.1
!
dial-peer cor custom
!
alias exec ship show ip int br | incl
alias exec mlslook show mls cef lookup
alias exec mlsentry show mls adj entry
alias exec reboot reload netboot  tftp://223.255.254.254/pradilla/s72033-pk9sv-mz
!
line con 0
 exec-timeout 0 0
line vty 0 4
 no login
!
scheduler runtime netinput 300
end
```

IPSec Stateful Failover Using HSRP and SSP Examples

The following two examples show IPSec stateful failover configurations using HSRP and SSP; one shows the configuration of the active chassis, the other the configuration of the standby chassis:

- IPSec Stateful Failover Using HSRP and SSP for the Active Chassis Configuration Example, page 28-36
- IPSec Stateful Failover Using HSRP and SSP for the Standby Chassis Configuration Example, page 28-38

Note The IPSec Stateful Failover Using HSRP and SSP feature is not supported in Cisco IOS Release 12.2(33)SRA.

Note These configuration examples do not protect the SSP traffic. To protect the SSP traffic, you will need to define a new crypto map and attach it to the SSP interface without the "ssp" tag. The ACL for this crypto map can be derived from the remote IP address and the TCP port that are defined in the SSP group.

IPSec Stateful Failover Using HSRP and SSP for the Active Chassis Configuration Example

The following example shows the configuration for an active chassis that is configured for an IPSec stateful failover using HSRP and SSP:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Active
!
redundancy
 main-cpu
  auto-sync standard
ip subnet-zero
!
no ip domain-lookup
!
ssp group 100
 remote 40.0.0.2
 redundancy KNIGHTSOFNI
no mls ip multicast aggregate
no mls ip multicast non-rpf cef
!
crypto isakmp policy 1
 encr 3des
 authentication pre-share
crypto isakmp key NEEWOMM address 0.0.0.0 0.0.0.0
crypto isakmp ssp 100
!
crypto ipsec security-association lifetime seconds 86400
!
crypto ipsec transform-set TS1 esp-3des esp-sha-hmac
!
crypto map ha ha replay-interval inbound 1000 outbound 1
crypto map ha 10 ipsec-isakmp
 set peer 172.16.31.3
 set transform-set TS1
 match address 101
!
spanning-tree extend system-id
no spanning-tree vlan 4
!
interface GigabitEthernet1/1
 no ip address
 no ip redirects
 crypto connect vlan 4
!
interface GigabitEthernet1/2
```

```
 ip address 40.0.0.1 255.255.255.0
 no ip redirects
 standby delay minimum 35 reload 60
 standby ip 40.0.0.100
 standby timers 3 5
 standby preempt
 standby track GigabitEthernet1/1
  standby track vlan 4
!
interface GigabitEthernet5/1/1
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,4,1002-1005
 switchport mode trunk
 flowcontrol receive on
 cdp enable
!
interface GigabitEthernet5/1/2
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 flowcontrol receive on
 cdp enable
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan4
 ip address 172.16.31.1 255.255.255.0
 standby delay minimum 35 reload 60
 standby ip 172.16.31.100
 standby timers 3 5
 standby preempt
 standby name KNIGHTSOFNI
 standby track GigabitEthernet1/1
 standby track GigabitEthernet1/2
 crypto map ha ssp 100
 crypto engine subslot 5/1
!
ip classless
ip route 10.11.1.1 255.255.255.255 172.16.31.3
no ip http server
ip pim bidir-enable
!
access-list 101 permit ip host 40.0.0.3 host 10.11.1.1
arp 127.0.0.12 0000.2100.0000 ARPA
!
line con 0
line vty 0 4
 login
 transport input lat pad mop telnet rlogin udptn nasi ssh
!
end
```

IPSec Stateful Failover Using HSRP and SSP for the Standby Chassis Configuration Example

The following example shows the configuration for a standby chassis that is configured for IPSec stateful failover using HSRP and SSP:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname StandBy
!
redundancy
 main-cpu
  auto-sync standard
ip subnet-zero
!
no ip domain-lookup
!
ssp group 100
 remote 40.0.0.1
 redundancy KNIGHTSOFNI
no mls ip multicast aggregate
no mls ip multicast non-rpf cef
!
crypto isakmp policy 1
 encr 3des
 authentication pre-share
crypto isakmp key NEEWOMM address 0.0.0.0 0.0.0.0
crypto isakmp ssp 100
!
crypto ipsec security-association lifetime seconds 86400
!
crypto ipsec transform-set TS1 esp-3des esp-sha-hmac
!
crypto map ha ha replay-interval inbound 1000 outbound 1
crypto map ha 10 ipsec-isakmp
 set peer 172.16.31.3
 set transform-set TS1
 match address 101
!
spanning-tree extend system-id
no spanning-tree vlan 4
!
interface GigabitEthernet1/1
 no ip address
 no ip redirects
 crypto connect vlan 4
!
interface GigabitEthernet1/2
 ip address 40.0.0.2 255.255.255.0
 no ip redirects
 standby delay minimum 35 reload 60
 standby ip 40.0.0.100
 standby timers 3 5
 standby preempt
 standby track GigabitEthernet1/1
  standby track vlan 4
!
interface GigabitEthernet5/1/1
 mtu 4500
 no ip address
 snmp trap link-status
```

```
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,4,1002-1005
 switchport mode trunk
 flowcontrol receive on
 cdp enable
!
interface GigabitEthernet5/1/2
 mtu 4500
 no ip address
 snmp trap link-status
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 1,1002-1005
 switchport mode trunk
 flowcontrol receive on
 cdp enable
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan4
 ip address 172.16.31.2 255.255.255.0
 standby delay minimum 35 reload 60
 standby ip 172.16.31.100
 standby timers 3 5
 standby preempt
 standby name KNIGHTSOFNI
 standby track GigabitEthernet1/1
 standby track GigabitEthernet1/2
 crypto map ha ssp 100
 crypto engine sublot 5/1
!
ip classless
ip route 10.11.1.1 255.255.255.255 172.16.31.3
no ip http server
ip pim bidir-enable
!
access-list 101 permit ip host 40.0.0.3 host 10.11.1.1
arp 127.0.0.12 0000.2100.0000 ARPA
!
line con 0
line vty 0 4
 login
 transport input lat pad mop telnet rlogin udptn nasi ssh
!
end
```

IPSec Stateful Failover Using a Blade Failure Group Configuration Example

The following example shows how to configure IPSec stateful failover using a Blade Failure Group (BFG):

```
Router(config)# redundancy
Router(config-red)# line-card-group 1 feature-card
Router(config-r-lc)# subslot 3/1
Router(config-r-lc)# subslot 5/1
```

CHAPTER 29

Configuring Monitoring and Accounting for the IPSec VPN SPA

This chapter provides information about configuring monitoring and accounting using the IPSec VPN SPA on the Cisco 7600 series router. It includes the following sections:

Note For detailed information on Cisco IOS IPSec cryptographic operations and policies, refer to the *Cisco IOS Security Configuration Guide* and *Cisco IOS Security Command Reference*.

For information about managing your system images and configuration files, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide* and *Cisco IOS Configuration Fundamentals Command Reference* publications.

For more information about the commands used in this chapter, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*. Also refer to the related Cisco IOS Release 12.2 software command reference and master index publications. For more information, see the "Related Documentation" section on page l.

Tip To ensure a successful configuration of your VPN using the IPSec VPN SPA, read all of the configuration summaries and guidelines before you perform any configuration tasks.

Overview of Monitoring and Accounting for the IPSec VPN SPA

This chapter describes some IPSec features that can be used to monitor and manage the IPSec VPN. These features include:

- The IPSec VPN Monitoring feature, which provides VPN session monitoring enhancements that will allow you to troubleshoot the Virtual Private Network (VPN) and monitor the end-user interface.
- The IPSec VPN Accounting feature, which enables session accounting records to be generated by indicating when the session starts and when it stops.
- The IPSec and IKE MIB Support for Cisco VRF-Aware IPSec feature, which provides manageability of Virtual Private Network routing and forwarding- (VRF-) aware IP security (IPSec) using MIBs.

Monitoring and Managing IPSec VPN Sessions

The IPSec VPN Monitoring feature provides VPN session monitoring enhancements that will allow you to troubleshoot the Virtual Private Network (VPN) and monitor the end-user interface. A crypto session is a set of IPSec connections (flows) between two crypto endpoints. If the two crypto endpoints use IKE as the keying protocol, they are IKE peers to each other. Typically, a crypto session consists of one IKE security association (for control traffic) and at least two IPSec security associations (for data traffic—one per each direction). There may be duplicated IKE security associations (SAs) and IPSec SAs or duplicated IKE SAs or IPSec SAs for the same session in the duration of rekeying or because of simultaneous setup requests from both sides.

Session monitoring enhancements include the following:

- Ability to specify an Internet Key Exchange (IKE) peer description in the configuration file
- Summary listing of crypto session status
- Syslog notification for crypto session up or down status
- Ability to clear both IKE and IP Security (IPSec) security associations (SAs) using one command-line interface (CLI)

Monitoring and Managing IPSec VPN Sessions Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring IPSec VPN Monitoring:

- You must be running Cisco IOS k8 or k9 crypto images on your router.

Adding the Description of an IKE Peer

To add the description of an IKE peer to an IPSec VPN session, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **crypto isakmp peer** {**ip-address** *ip-address*}	Enables an IPSec peer for IKE querying of authentication, authorization, and accounting (AAA) for tunnel attributes in aggressive mode and enters ISAKMP peer configuration mode. • *ip-address*—Integer specifying number of minutes.
Step 2	Router(config-isakmp-peer)# **description** *description*	Adds a description for an IKE peer. • description—Description identifying the peer.

Verifying Peer Descriptions

To verify peer descriptions, enter the **show crypto isakmp peer** command:

```
Router# show crypto isakmp peer

Peer: 10.2.2.9 Port: 500
Description: connection from site A
flags: PEER_POLICY
```

When the peer at address 10.2.2.9 connects and the session comes up, the syslog status will be shown as follows:

```
%CRYPTO-5-SESSION_STATUS: Crypto tunnel is UP. Peer 10.2.2.9:500 Description: connection
from site A Id: ezvpn
```

Getting a Summary Listing of Crypto Session Status

You can get a list of all the active VPN sessions by entering the **show crypto session** command. The listing will include the following:

- Interface
- IKE peer description, if available
- IKE SAs that are associated with the peer by which the IPSec SAs are created
- IPSec SAs serving the flows of a session

Multiple IKE or IPSec SAs may be established for the same peer (for the same session), in which case IKE peer descriptions will be repeated with different values for the IKE SAs that are associated with the peer and for the IPSec SAs that are serving the flows of the session.

You can also use the **show crypto session detail** variant of this command to obtain more detailed information about the sessions.

The following is sample output for the **show crypto session** output without the **detail** keyword:

```
Router# show crypto session

Crypto session current status

Interface: FastEthernet0/1
Session status: UP-ACTIVE
Peer: 172.0.0.2/500
IKE SA: local 172.0.0.1/500 remote 172.0.0.2/500 Active
IPSEC FLOW: permit ip 10.10.10.0/255.255.255.0 10.30.30.0/255.255.255.0
Active SAs: 2, origin: crypto map
```

The following is sample output using the **show crypto sessio**n command and the **detail** keyword:

```
Router# show crypto session detail

Interface: Tunnel0
Session status: UP-ACTIVE
Peer: 10.1.1.3 port 500 fvrf: (none) ivrf: (none)
Desc: this is my peer at 10.1.1.3:500 Green
Phase1_id: 10.1.1.3
IKE SA: local 10.1.1.4/500 remote 10.1.1.3/500 Active
Capabilities:(none) connid:3 lifetime:22:03:24
IPSEC FLOW: permit 47 host 10.1.1.4 host 10.1.1.3
Active SAs: 0, origin: crypto map
Inbound: #pkts dec'ed 0 drop 0 life (KB/Sec) 0/0
Outbound: #pkts enc'ed 0 drop 0 life (KB/Sec) 0/0
IPSEC FLOW: permit ip host 10.1.1.4 host 10.1.1.3
Active SAs: 4, origin: crypto map
Inbound: #pkts dec'ed 4 drop 0 life (KB/Sec) 4605665/2949
Outbound: #pkts enc'ed 4 drop 1 life (KB/Sec) 4605665/2949
```

Syslog Notification for Crypto Session Up or Down Status

The Syslog Notification for Crypto Session Up or Down Status function provides syslog notification every time the crypto session comes up or goes down.

The following is a sample syslog notification showing that a crypto session is up:

```
%CRYPTO-5-SESSION_STATUS: Crypto session is UP. Peer 10.6.6.1:500 fvrf=name10 ivrf=name20
Description: SJC24-2-VPN-Gateway Id: 10.5.5.2
```

The following is a sample syslog notification showing that a crypto session is down:

```
%CRYPTO-5-SESSION_STATUS: Crypto session is DOWN. Peer 10.6.6.1:500 fvrf=name10
ivrf=name20 Description: SJC24-2-VPN-Gateway Id: 10.5.5.2
```

Clearing a Crypto Session

In previous Cisco IOS versions, there was no single command to clear both IKE and IPSec connections (that is, SAs). Instead, you had to use the **clear crypto isakmp** command to clear IKE and the **clear crypto ipsec** command to clear IPSec. The **clear crypto session** command allows you to clear both IKE and IPSec with a single command. To clear a specific crypto session or a subset of all the sessions (for example, a single tunnel to one remote site), you need to provide session-specific parameters, such as a local or remote IP address, a local or remote port, a front door VRF (FVRF) name, or an inside VRF (IVRF) name. Typically, the remote IP address will be used to specify a single tunnel to be deleted.

If a local IP address is provided as a parameter when you use the **clear crypto session** command, all the sessions (and their IKE SAs and IPSec SAs) that share the IP address as a local crypto endpoint (IKE local address) will be cleared. If you do not provide a parameter when you use the **clear crypto session** command, all IPSec SAs and IKE SAs that are in the router will be deleted.

To clear a crypto session, use the **clear crypto session** command in privileged EXEC mode from the router command line. No configuration statements are required in the configuration file to use this command.

```
Router# clear crypto session
```

For complete configuration information for IPSec VPN Monitoring, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_4/gt_ipsvm.htm

For a Monitoring and Managing VPN Sessions configuration example, see the "Monitoring and Managing VPN Sessions Configuration Example" section on page 29-11.

Configuring IPSec VPN Accounting

The IPSec VPN Accounting feature enables session accounting records to be generated by indicating when the session starts and when it stops.

A VPN session is defined as an Internet Key Exchange (IKE) security association (SA) and the one or more SA pairs that are created by the IKE SA. The session starts when the first IP Security (IPSec) pair is created and stops when all IPSec SAs are deleted. If IPSec accounting is configured, after IKE phases are complete, an accounting start record is generated for the session. New accounting records are not generated during a rekeying.

Session-identifying information and session-usage information is passed to the Remote Authentication Dial-In User Service (RADIUS) server via standard RADIUS attributes and vendor-specific attributes (VSAs).

To enable IPSec VPN Accounting, perform the following steps beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **aaa new-model**	Enables periodic interim accounting records to be sent to the accounting server.
Step 2	Router(config)# **aaa authentication login** *list-name* **group radius**	Sets authentication, authorization, and accounting (AAA) authentication at login using RADIUS servers. • *list-name*—Character string used to name the list of authentication methods activated when a user logs in. • **group radius**—Uses the list of all RADIUS servers for authentication.

	Command	Purpose
Step 3	Router(config)# **aaa authorization network** *list-name* **group radius**	Runs authorization for all network-related service requests, including Serial Line Internet Protocol (SLIP), PPP, PPP Network Control Programs (NCPs), and AppleTalk Remote Access (ARA). • *list-name*—Character string used to name the list of authorization methods activated when a user logs in. • **group radius**—Uses the list of all RADIUS servers for authentication.
Step 4	Router(config)# **aaa accounting network** *list-name* **start-stop** [**broadcast**] **group radius**	Enables AAA accounting of network-related requested services for billing or security purposes when you use RADIUS. • *list-name*—Character string used to name the list of the accounting methods. • **start-stop**—Sends a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record is sent in the background. The requested user process begins regardless of whether the "start" accounting notice was received by the accounting server. • **broadcast**—(Optional) Enables sending accounting records to multiple AAA servers. Simultaneously sends accounting records to the first server in each group. If the first server is unavailable, failover occurs using the backup servers defined within that group. • **group radius**—Uses the list of all RADIUS servers for authentication as defined by the **aaa group server radius** command.
Step 5	Router(config)# **aaa session-id common**	Specifies whether the same session ID will be used for each AAA accounting service type within a call or whether a different session ID will be assigned to each accounting service type. • **common**—Ensures that all session identification (ID) information that is sent out for a given call will be made identical. The default behavior is common.
Step 6	Router(config)# **crypto isakmp profile** *profile-name*	Audits IP security (IPSec) user sessions and enters isakmp-profile configuration mode. • *profile-name*—Name of the user profile. To associate a user profile with the RADIUS server, the user profile name must be identified.

	Command	Purpose
Step 7	Router(conf-isa-prof)# **vrf** *ivrf*	Associates the on-demand address pool with a Virtual Private Network (VPN) routing and forwarding (VRF) instance name. • *ivrf*—VRF to which the IPSec tunnel will be mapped.
Step 8	Router(conf-isa-prof)# **match identity group** *group-name*	Matches an identity from a peer in an ISAKMP profile. • *group-name*—A Unity group that matches identification (ID) type ID_KEY_ID. If Unity and main mode Rivest, Shamir, and Adelman (RSA) signatures are used, the *group-name* argument matches the Organizational Unit (OU) field of the Distinguished Name (DN).
Step 9	Router(conf-isa-prof)# **client authentication list** *list-name*	Configures Internet Key Exchange (IKE) extended authentication (XAUTH) in an Internet Security Association and Key Management Protocol (ISAKMP) profile. • *list-name*—Character string used to name the list of authentication methods activated when a user logs in. The list name must match the list name that was defined during the authentication, authorization, and accounting (AAA) configuration.
Step 10	Router(conf-isa-prof)# **isakmp authorization list** *list-name*	Configures an IKE shared secret and other parameters using the AAA server in an ISAKMP profile. The shared secret and other parameters are generally pushed to the remote peer via mode configuration (MODECFG). • *list-name*—AAA authorization list used for configuration mode attributes or preshared keys for aggressive mode.
Step 11	Router(conf-isa-prof)# **client configuration address** [**initiate** | **respond**]	Configures IKE mode configuration (MODECFG) in the ISAKMP profile. • **initiate**—Router will attempt to set IP addresses for each peer. • **respond**—Router will accept requests for IP addresses from any requesting peer.
Step 12	Router(conf-isa-prof)# **accounting** *list-name*	Enables AAA accounting services for all peers that connect via this ISAKMP profile. • *list-name*—Name of a client accounting list.
Step 13	Router(conf-isa-prof)# **exit**	Exits isakmp-profile configuration mode and enters global configuration mode.

	Command	Purpose
Step 14	Router(config)# **crypto dynamic-map** *dynamic-map-name dynamic-seq-num*	Creates a dynamic crypto map template and enters the crypto map configuration command mode. • *dynamic-map-name*—Name of the dynamic crypto map set that should be used as the policy template. • *dynamic-seq-num*—Sequence number you assign to the dynamic crypto map entry.
Step 15	Router(config-crypto-map)# **set transform-set** *transform-set-name*	Specifies which transform sets can be used with the crypto map template. A transform set defines IPSec security protocols and algorithms. Transform sets and their accepted values are described in the *Cisco IOS Security Command Reference*. • *transform-set-name*—Name of the transform set.
Step 16	Router(config-crypto-map)# **set isakmp-profile** *profile-name*	Sets the ISAKMP profile name. • *profile-name*—Name of the ISAKMP profile.
Step 17	Router(config-crypto-map)# **reverse-route** [**remote-peer**]	Allows routes (IP addresses) to be injected for destinations behind the VPN remote tunnel endpoint and may include a route to the tunnel endpoint itself (using the **remote-peer** keyword for the crypto map). • **remote-peer**—(Optional) Routes of public IP addresses and IP security (IPSec) tunnel destination addresses are inserted into the routing table.
Step 18	Router(config-crypto-map)# **exit**	Exits crypto map configuration mode and enters global configuration mode.
Step 19	Router(config)# **crypto map** *map-name* **ipsec-isakmp dynamic** *dynamic-map-name*	Creates a crypto profile that provides a template for configuration of dynamically created crypto maps. • *map-nam*e—Name that identifies the crypto map set. • *dynamic-map-name*—Name of the dynamic crypto map set that should be used as the policy template.

	Command	Purpose
Step 20	Router(config)# **radius-server host** *ip-address* [**auth-port** *auth-port-number*] [**acct-port** *acct-port-number*]	Specifies a RADIUS server host. • *ip-address* —IP address of the RADIUS server host. • **auth-port** *auth-port-number*—(Optional) UDP destination port number for authentication requests; the host is not used for authentication if set to 0. If unspecified, the port number defaults to 1645. • **acct-port** *acct-port-number*—(Optional) UDP destination port number for accounting requests; the host is not used for accounting if set to 0. If unspecified, the port number defaults to 1646.
Step 21	Router(config)# **radius-server key** *string*	Sets the authentication and encryption key for all RADIUS communications between the router and the RADIUS daemon. • *string*—The unencrypted (cleartext) shared key.
Step 22	Router(config)# **interface** *type slot/subslot/port*	Configures an interface type and enters interface configuration mode. • type—Specifies the type of interface. • *slot*—Specifies the chassis slot number where the SIP is installed. • *subslot*—Specifies the secondary slot number on a SIP where a SPA is installed. • *port*—Specifies the number of the interface port on the SPA
Step 23	Router(config-if)# **crypto map** *map-name*	Applies a previously defined crypto map set to an interface. • *map-name*—Name that identifies the crypto map set.

Configuring Accounting Updates

To send accounting updates while a session is "up," enter the following optional **aaa accounting update periodic command** in global configuration mode:

Router(config)# **aaa accounting update periodic** *number*

In this command, *number* is an integer specifying the interval (in number of minutes) at which accounting records are to be sent to the accounting server.

For complete configuration information for IPSec VPN Accounting, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t15/ft_evpna.htm

For IPSec VPN accounting configuration examples, see the "IPSec VPN Accounting Configuration Examples" section on page 29-11.

Configuring IPSec and IKE MIB Support for Cisco VRF-Aware IPSec

The IPSec and IKE MIB Support for Cisco VRF-Aware IPSec feature provides manageability of Virtual Private Network routing and forwarding (VRF)-aware IP security (IPSec) using MIBs. The benefit of this feature is that VRF-aware IPSec MIBs provide the granular details of IPSec statistics and performance metrics on a VRF basis.

Note The IPSec and IKE MIB Support for the Cisco VRF-Aware IPSec feature is only supported as of Cisco IOS Release 12.2(33)SRA.

MIBs Supported by the IPSec and IKE MIB Support for Cisco VRF-Aware IPSec Feature

The following MIBs are supported by the IPSec and IKE MIB Support for the Cisco VRF-Aware IPSec feature:

- CISCO-IPSEC-FLOW-MONITOR-MIB
- ISCO-IPSEC-MIB
- The CISCO-IPSEC-POLICY-MAP-MIB continues to be supported. However, because this MIB applies to the entire router rather than to a specific VPN VRF instance, it is not VRF-aware; therefore, polling of the object identifiers (OIDs) that belong to this MIB is accomplished with respect to the global VRF context.

Configuring IPSec and IKE MIB Support for Cisco VRF-Aware IPsec

No special configuration is needed for this feature. The SNMP framework can be used to manage VRF-aware IPSec using MIBs.

For complete information for IPSec and IKE MIB Support for Cisco VRF-Aware IPSec, refer to this URL:

http://www.cisco.com/en/US/products/ps6441/products_feature_guide09186a00804ff67b.html

For a sample SNMP configuration example, see the "IPSec and IKE MIB Support for Cisco VRF-Aware IPSec Configuration Example" section on page 29-11.

Configuration Examples

This section provides examples of the following configurations:

Monitoring and Managing VPN Sessions Configuration Example

The following example shows how to configure an IKE peer for IPSec VPN monitoring:

```
Router(config)# crypto isakmp peer address 10.2.2.9
Router(config-isakmp-peer)# description connection from site
```

IPSec VPN Accounting Configuration Examples

The following examples show how to configure IPSec VPN accounting initially and how to configure accounting updates:

- IPSec VPN Accounting Configuration Example, page 29-11
- IPSec VPN Accounting Updates Configuration Example, page 29-11

IPSec VPN Accounting Configuration Example

The following example shows how to enable the IPSec VPN accounting feature:

```
Router(config)# aaa new-model
Router(config)# aaa authentication login cisco-client group radius
Router(config)# aaa authorization network cisco-client group radius
Router(config)# aaa accounting network acc start-stop broadcast group radius
Router(config)# aaa session-id common
Router(config)# crypto isakmp profile cisco
Router(conf-isa-prof)# vrf cisco
Router(conf-isa-prof)# match identity group cisco
Router(conf-isa-prof)# client authentication list cisco
Router(conf-isa-prof)# isakmp authorization list cisco-client
Router(conf-isa-prof)# client configuration address respond
Router(conf-isa-prof)# accounting acc
Router(conf-isa-prof)# exit
Router(config)# crypto dynamic-map mymap 10 ipsec-isakmp
Router(config-crypto-map)# set transform-set aswan
Router(config-crypto-map)# set isakmp-profile cisco
Router(config-crypto-map)# reverse-route
Router(config-crypto-map)# exit
Router(config)# crypto map mymap ipsec-isakmp dynamic dmap
Router(config)# radius-server host 172.16.1.4
Router(config)# radius-server key nsite
Router(config)# interface FastEthernet 1/0
Router(config-if)# crypto map mymap
```

IPSec VPN Accounting Updates Configuration Example

The following example shows how to send updates while a session is “up”:

```
Router(config)# aaa accounting update periodic 1-2147483647
```

IPSec and IKE MIB Support for Cisco VRF-Aware IPSec Configuration Example

The following SNMP example is for a typical hub configuration that has two VRFs. The output is what you would see if you were to poll for the IPSec security association (SA). Router 3745b is the VRF-aware router.

The following output shows that two VRFs have been configured (vrf1 and vrf2):

Two VRFs Configured

```
Router3745b# show running-config
Building configuration...
Current configuration : 6567 bytes
!
version 12.4
service timestamps debug datetime msec localtime
service timestamps log uptime
no service password-encryption
!
hostname ipsecf-3745b
!
boot-start-marker
boot-end-marker
!
no logging console
enable password lab
!
no aaa new-model
!
resource policy
!
memory-size iomem 5
clock timezone PST -8
clock summer-time PDT recurring
ip subnet-zero
ip cef
!
ip vrf vrf1
rd 1:101
context vrf-vrf1-context
route-target export 1:101
route-target import 1:101
!
ip vrf vrf2
rd 2:101
context vrf-vrf2-context
route-target export 2:101
route-target import 2:101
!
no ip domain lookup
!
crypto keyring vrf1-1 vrf vrf1
pre-shared-key address 10.1.1.1 255.255.255.0 key vrf1-1
crypto keyring vrf2-1 vrf vrf2
pre-shared-key address 10.1.2.1 255.255.255.0 key vrf2-1
!
crypto isakmp policy 1
authentication pre-share
!
crypto isakmp policy 50
authentication pre-share
crypto isakmp key global1-1 address 10.1.151.1
crypto isakmp key global2-1 address 10.1.152.1
crypto isakmp profile vrf1-1
keyring vrf1-1
match identity address 10.1.1.1 255.255.255.255 vrf1
crypto isakmp profile vrf2-1
keyring vrf2-1
match identity address 10.1.2.1 255.255.255.255 vrf2
!
crypto ipsec security-association lifetime kilobytes 99000
crypto ipsec security-association lifetime seconds 5000
```

```
!
crypto ipsec transform-set tset ah-sha-hmac esp-des esp-sha-hmac
!
crypto map global1-1 10 ipsec-isakmp
set peer 10.1.151.1
set transform-set tset
match address 151
!
crypto map global2-1 10 ipsec-isakmp
set peer 10.1.152.1
set transform-set tset
match address 152
!
crypto map vrf1-1 10 ipsec-isakmp
set peer 10.1.1.1
set transform-set tset
set isakmp-profile vrf1-1
match address 101
!
crypto map vrf2-1 10 ipsec-isakmp
set peer 10.1.2.1
set transform-set tset
set isakmp-profile vrf2-1
match address 102
!
interface FastEthernet0/0
ip address 10.1.38.25 255.255.255.0
no ip mroute-cache
duplex auto
speed auto
!
interface Serial0/0
no ip address
shutdown
clock rate 2000000
!
interface FastEthernet0/1
no ip address
no ip mroute-cache
shutdown
duplex auto
speed auto
!
interface Serial0/1
no ip address
shutdown
clock rate 2000000
!
interface Serial1/0
no ip address
encapsulation frame-relay
no ip route-cache cef
no ip route-cache
no ip mroute-cache
no keepalive
serial restart-delay 0
clock rate 128000
no frame-relay inverse-arp
!
interface Serial1/0.1 point-to-point
ip vrf forwarding vrf1
ip address 10.3.1.1 255.255.255.0
no ip route-cache
frame-relay interface-dlci 21
```

```
!
interface Serial1/0.2 point-to-point
ip vrf forwarding vrf2
ip address 10.3.2.1 255.255.255.0
no ip route-cache
frame-relay interface-dlci 22
!
interface Serial1/0.151 point-to-point
ip address 10.7.151.1 255.255.255.0
no ip route-cache
frame-relay interface-dlci 151
!
interface Serial1/0.152 point-to-point
ip address 10.7.152.1 255.255.255.0
no ip route-cache
frame-relay interface-dlci 152
!
interface Serial1/1
no ip address
no ip mroute-cache
shutdown
serial restart-delay 0
!
interface Serial1/2
no ip address
encapsulation frame-relay
no ip route-cache cef
no ip route-cache
no ip mroute-cache
no keepalive
serial restart-delay 0
no frame-relay inverse-arp
!
interface Serial1/2.1 point-to-point
ip vrf forwarding vrf1
ip address 10.1.1.2 255.255.255.0
no ip route-cache
frame-relay interface-dlci 21
crypto map vrf1-1
!
interface Serial1/2.2 point-to-point
ip vrf forwarding vrf2
ip address 10.1.2.2 255.255.255.0
no ip route-cache
frame-relay interface-dlci 22
crypto map vrf2-1
!
interface Serial1/2.151 point-to-point
ip address 10.5.151.2 255.255.255.0
no ip route-cache
frame-relay interface-dlci 151
crypto map global1-1
!
interface Serial1/2.152 point-to-point
ip address 10.5.152.2 255.255.255.0
no ip route-cache
frame-relay interface-dlci 152
crypto map global2-1
!
interface Serial1/3
no ip address
no ip mroute-cache
shutdown
serial restart-delay 0
```

```
!
ip default-gateway 10.1.38.1
ip classless
ip route 10.1.1.6 255.255.255.255 10.1.151.1
ip route 10.2.1.6 255.255.255.255 10.1.152.1
ip route 10.6.2.1 255.255.255.255 10.7.151.2
ip route 10.6.2.2 255.255.255.255 10.7.152.2
ip route 172.19.216.110 255.255.255.255 FastEthernet0/0
ip route vrf vrf1 10.20.1.1 255.255.255.255 10.1.1.1
ip route vrf vrf1 10.22.1.1 255.255.255.255 10.30.1.1
ip route vrf vrf2 10.20.2.1 255.255.255.255 10.1.2.1
ip route vrf vrf2 10.22.2.1 255.255.255.255 10.30.1.2
!
ip http server
no ip http secure-server
!
ip access-list standard vrf-vrf1-context
ip access-list standard vrf-vrf2-context
!
access-list 101 permit ip host 10.22.1.1 host 10.20.1.1
access-list 102 permit ip host 10.22.2.1 host 10.20.2.1
access-list 151 permit ip host 10.6.2.1 host 10.1.1.6
access-list 152 permit ip host 10.6.2.2 host 10.2.1.6
snmp-server group abc1 v2c context vrf-vrf1-context read view_vrf1 notify
*tv.FFFFFFFF.FFFFFFFF.FFFFFFFF.F access vrf-vrf1-context
snmp-server group abc2 v2c context vrf-vrf2-context read view_vrf2 notify
*tv.FFFFFFFF.FFFFFFFF.FFFFFFFF.F access vrf-vrf2-context
snmp-server view view_vrf1 iso included
snmp-server view view_vrf2 iso included
snmp-server community abc1 RW
snmp-server community global1 RW
snmp-server community abc2 RW
snmp-server community global2 RW
snmp-server enable traps tty
snmp-server enable traps config
snmp-server host 172.19.216.110 version 2c abc1
snmp-server host 172.19.216.110 vrf vrf1 version 2c abc1 udp-port 2001  ipsec isakmp
snmp-server host 172.19.216.110 version 2c abc2
snmp-server host 172.19.216.110 vrf vrf2 version 2c abc2 udp-port 2002  ipsec isakmp
snmp-server context vrf-vrf1-context
snmp-server context vrf-vrf2-context
!
snmp mib community-map  abc1 context vrf-vrf1-context
snmp mib community-map  abc2 context vrf-vrf2-context
!
control-plane
!
line con 0
exec-timeout 0 0
line aux 0
line vty 0 4
login
!
webvpn context Default_context
ssl authenticate verify all
!
no inservice
!
end
```

Both VRFs Cleared

The following output, for abc1 and abc2, shows that both VRFs have been "cleared" to ensure that all the counters are initialized to a known value.

The following output shows that VRF abc1 has been cleared:

```
orcas:2> setenv SR_MGR_CONF /users/green1
orcas:3> setenv SR_UTIL_SNMP_VERSION v2c
orcas:5> setenv SR_UTIL_COMMUNITY abc1
orcas:6> setenv SR_MGR_CONF_DIR /users/green1
orcas:7> /auto/sw/packages/snmpr/10.14.2.0/solaris2bin/getmany -v2c 10.1.38.25
cipSecMIBObjects
cipSecMibLevel.0 = 1
cikeGlobalActiveTunnels.0 = 0
cikeGlobalPreviousTunnels.0 = 0
cikeGlobalInOctets.0 = 0
cikeGlobalInPkts.0 = 0
cikeGlobalInDropPkts.0 = 0
cikeGlobalInNotifys.0 = 0
cikeGlobalInP2Exchgs.0 = 0
cikeGlobalInP2ExchgInvalids.0 = 0
cikeGlobalInP2ExchgRejects.0 = 0
cikeGlobalInP2SaDelRequests.0 = 0
cikeGlobalOutOctets.0 = 0
cikeGlobalOutPkts.0 = 0
cikeGlobalOutDropPkts.0 = 0
cikeGlobalOutNotifys.0 = 0
cikeGlobalOutP2Exchgs.0 = 0
cikeGlobalOutP2ExchgInvalids.0 = 0
cikeGlobalOutP2ExchgRejects.0 = 0
cikeGlobalOutP2SaDelRequests.0 = 0
cikeGlobalInitTunnels.0 = 0
cikeGlobalInitTunnelFails.0 = 0
cikeGlobalRespTunnelFails.0 = 0
cikeGlobalSysCapFails.0 = 0
cikeGlobalAuthFails.0 = 0
cikeGlobalDecryptFails.0 = 0
cikeGlobalHashValidFails.0 = 0
cikeGlobalNoSaFails.0 = 0
cipSecGlobalActiveTunnels.0 = 0
cipSecGlobalPreviousTunnels.0 = 0
cipSecGlobalInOctets.0 = 0
cipSecGlobalHcInOctets.0 = 0x00
cipSecGlobalInOctWraps.0 = 0
cipSecGlobalInDecompOctets.0 = 0
cipSecGlobalHcInDecompOctets.0 = 0x00
cipSecGlobalInDecompOctWraps.0 = 0
cipSecGlobalInPkts.0 = 0
cipSecGlobalInDrops.0 = 0
cipSecGlobalInReplayDrops.0 = 0
cipSecGlobalInAuths.0 = 0
cipSecGlobalInAuthFails.0 = 0
cipSecGlobalInDecrypts.0 = 0
cipSecGlobalInDecryptFails.0 = 0
cipSecGlobalOutOctets.0 = 0
cipSecGlobalHcOutOctets.0 = 0x00
cipSecGlobalOutOctWraps.0 = 0
cipSecGlobalOutUncompOctets.0 = 0
cipSecGlobalHcOutUncompOctets.0 = 0x00
cipSecGlobalOutUncompOctWraps.0 = 0
cipSecGlobalOutPkts.0 = 0
cipSecGlobalOutDrops.0 = 0
cipSecGlobalOutAuths.0 = 0
cipSecGlobalOutAuthFails.0 = 0
```

```
cipSecGlobalOutEncrypts.0 = 0
cipSecGlobalOutEncryptFails.0 = 0
cipSecGlobalProtocolUseFails.0 = 0
ipSecGlobalNoSaFails.0 = 0
cipSecGlobalSysCapFails.0 = 0
cipSecHistTableSize.0 = 200
cipSecHistCheckPoint.0 = ready(1)
cipSecFailTableSize.0 = 200
cipSecTrapCntlIkeTunnelStart.0 = enabled(1)
cipSecTrapCntlIkeTunnelStop.0 = enabled(1)
cipSecTrapCntlIkeSysFailure.0 = disabled(2)
cipSecTrapCntlIkeCertCrlFailure.0 = disabled(2)
cipSecTrapCntlIkeProtocolFail.0 = disabled(2)
cipSecTrapCntlIkeNoSa.0 = disabled(2)
cipSecTrapCntlIpSecTunnelStart.0 = enabled(1)
cipSecTrapCntlIpSecTunnelStop.0 = enabled(1)
cipSecTrapCntlIpSecSysFailure.0 = disabled(2)
cipSecTrapCntlIpSecSetUpFailure.0 = disabled(2)
cipSecTrapCntlIpSecEarlyTunTerm.0 = disabled(2)
cipSecTrapCntlIpSecProtocolFail.0 = disabled(2)
cipSecTrapCntlIpSecNoSa.0 = disabled(2)
```

The following output shows that VRF abc2 has been cleared:

```
orcas:8> setenv SR_UTIL_COMMUNITY abc2
orcas:9> /auto/sw/packages/snmpr/14.2.0.0/solaris2bin/getmany -v2c 10.1.38.25
cipSecMIBObjects
cipSecMibLevel.0 = 1
cikeGlobalActiveTunnels.0 = 0
cikeGlobalPreviousTunnels.0 = 0
cikeGlobalInOctets.0 = 0
cikeGlobalInPkts.0 = 0
cikeGlobalInDropPkts.0 = 0
cikeGlobalInNotifys.0 = 0
cikeGlobalInP2Exchgs.0 = 0
cikeGlobalInP2ExchgInvalids.0 = 0
cikeGlobalInP2ExchgRejects.0 = 0
cikeGlobalInP2SaDelRequests.0 = 0
cikeGlobalOutOctets.0 = 0
cikeGlobalOutPkts.0 = 0
cikeGlobalOutDropPkts.0 = 0
cikeGlobalOutNotifys.0 = 0
cikeGlobalOutP2Exchgs.0 = 0
cikeGlobalOutP2ExchgInvalids.0 = 0
cikeGlobalOutP2ExchgRejects.0 = 0
cikeGlobalOutP2SaDelRequests.0 = 0
cikeGlobalInitTunnels.0 = 0
cikeGlobalInitTunnelFails.0 = 0
cikeGlobalRespTunnelFails.0 = 0
cikeGlobalSysCapFails.0 = 0
cikeGlobalAuthFails.0 = 0
cikeGlobalDecryptFails.0 = 0
cikeGlobalHashValidFails.0 = 0
cikeGlobalNoSaFails.0 = 0
cipSecGlobalActiveTunnels.0 = 0
cipSecGlobalPreviousTunnels.0 = 0
cipSecGlobalInOctets.0 = 0
cipSecGlobalHcInOctets.0 = 0x00
cipSecGlobalInOctWraps.0 = 0
cipSecGlobalInDecompOctets.0 = 0
cipSecGlobalHcInDecompOctets.0 = 0x00
cipSecGlobalInDecompOctWraps.0 = 0
cipSecGlobalInPkts.0 = 0
cipSecGlobalInDrops.0 = 0
```

```
cipSecGlobalInReplayDrops.0 = 0
cipSecGlobalInAuths.0 = 0
cipSecGlobalInAuthFails.0 = 0
cipSecGlobalInDecrypts.0 = 0
cipSecGlobalInDecryptFails.0 = 0
cipSecGlobalOutOctets.0 = 0
cipSecGlobalHcOutOctets.0 = 0x00
cipSecGlobalOutOctWraps.0 = 0
cipSecGlobalOutUncompOctets.0 = 0
cipSecGlobalHcOutUncompOctets.0 = 0x00
cipSecGlobalOutUncompOctWraps.0 = 0
cipSecGlobalOutPkts.0 = 0
cipSecGlobalOutDrops.0 = 0
cipSecGlobalOutAuths.0 = 0
cipSecGlobalOutAuthFails.0 = 0
cipSecGlobalOutEncrypts.0 = 0
cipSecGlobalOutEncryptFails.0 = 0
cipSecGlobalProtocolUseFails.0 = 0
cipSecGlobalNoSaFails.0 = 0
cipSecGlobalSysCapFails.0 = 0
cipSecHistTableSize.0 = 200
cipSecHistCheckPoint.0 = ready(1)
cipSecFailTableSize.0 = 200
cipSecTrapCntlIkeTunnelStart.0 = enabled(1)
cipSecTrapCntlIkeTunnelStop.0 = enabled(1)
cipSecTrapCntlIkeSysFailure.0 = disabled(2)
cipSecTrapCntlIkeCertCrlFailure.0 = disabled(2)
cipSecTrapCntlIkeProtocolFail.0 = disabled(2)
cipSecTrapCntlIkeNoSa.0 = disabled(2)
cipSecTrapCntlIpSecTunnelStart.0 = enabled(1)
cipSecTrapCntlIpSecTunnelStop.0 = enabled(1)
cipSecTrapCntlIpSecSysFailure.0 = disabled(2)
cipSecTrapCntlIpSecSetUpFailure.0 = disabled(2)
cipSecTrapCntlIpSecEarlyTunTerm.0 = disabled(2)
cipSecTrapCntlIpSecProtocolFail.0 = disabled(2)
cipSecTrapCntlIpSecNoSa.0 = disabled(2)
orcas:10>
orcas:10>
orcas:10>
```

VRF abc1 Pinged

The following output shows that VRF abc1 has been pinged:

```
Router3745a# ping
Protocol [ip]:
Target IP address: 10.22.1.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 10.20.1.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.22.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.20.1.1
```

VRF abc1 Polled

Polling VRF abc1 results in the following output:

Note After the ping, the counters should show some nonzero values.

```
orcas:10>
orcas:12> setenv SR_UTIL_COMMUNITY abc1
orcas:13> /auto/sw/packages/snmpr/10.14.2.0/solaris2bin/getmany -v2c 10.1.38.25
cipSecMIBObjects
cipSecMibLevel.0 = 1
cikeGlobalActiveTunnels.0 = 1
cikeGlobalPreviousTunnels.0 = 0
cikeGlobalInOctets.0 = 336
cikeGlobalInPkts.0 = 2
cikeGlobalInDropPkts.0 = 0
cikeGlobalInNotifys.0 = 1
cikeGlobalInP2Exchgs.0 = 2
cikeGlobalInP2ExchgInvalids.0 = 0
cikeGlobalInP2ExchgRejects.0 = 0
cikeGlobalInP2SaDelRequests.0 = 0
cikeGlobalOutOctets.0 = 344
cikeGlobalOutPkts.0 = 2
cikeGlobalOutDropPkts.0 = 0
cikeGlobalOutNotifys.0 = 0
cikeGlobalOutP2Exchgs.0 = 1
cikeGlobalOutP2ExchgInvalids.0 = 0
cikeGlobalOutP2ExchgRejects.0 = 0
cikeGlobalOutP2SaDelRequests.0 = 0
cikeGlobalInitTunnels.0 = 0
cikeGlobalInitTunnelFails.0 = 0
cikeGlobalRespTunnelFails.0 = 0
cikeGlobalSysCapFails.0 = 0
cikeGlobalAuthFails.0 = 0
cikeGlobalDecryptFails.0 = 0
cikeGlobalHashValidFails.0 = 0
cikeGlobalNoSaFails.0 = 0
cikePeerLocalAddr.1.15.48.49.48.46.48.48.49.46.48.48.49.46.48.48.50.1.15.48.49.48.46.48.48
.49.46.48.48.49.46.48.48.49.1 = 0a 01  01 02
cikePeerRemoteAddr.1.15.48.49.48.46.48.48.49.46.48.48.49.46.48.48.50.1.15.48.49.48.46.48.4
8.49.46.48.48.49.46.48.48.49.1 = 0a 01  01 01
cikePeerActiveTime.1.15.48.49.48.46.48.48.49.46.48.48.49.46.48.48.50.1.15.48.49.48.46.48.4
8.49.46.48.48.49.46.48.48.49.1 = 13743
cikePeerActiveTunnelIndex.1.15.48.49.48.46.48.48.49.46.48.48.49.46.48.48.50.1.15.48.49.48.
46.48.48.49.46.48.48.49.46.48.48.49.1 = 1
cikeTunLocalType.1 = ipAddrPeer(1)
cikeTunLocalValue.1 = 010.001.001.002
cikeTunLocalAddr.1 = 0a 01  01 02
cikeTunLocalName.1 = ipsecf-3745b
cikeTunRemoteType.1 = ipAddrPeer(1)
cikeTunRemoteValue.1 = 010.001.001.001
cikeTunRemoteAddr.1 = 0a 01  01 01
cikeTunRemoteName.1 =
cikeTunNegoMode.1 = main(1)
cikeTunDiffHellmanGrp.1 = dhGroup1(2)
cikeTunEncryptAlgo.1 = des(2)
cikeTunHashAlgo.1 = sha(3)
cikeTunAuthMethod.1 = preSharedKey(2)
cikeTunLifeTime.1 = 86400
cikeTunActiveTime.1 = 13752
cikeTunSaRefreshThreshold.1 = 0
cikeTunTotalRefreshes.1 = 0
```

```
cikeTunInOctets.1 = 336
cikeTunInPkts.1 = 2
cikeTunInDropPkts.1 = 0
cikeTunInNotifys.1 = 1
cikeTunInP2Exchgs.1 = 2
cikeTunInP2ExchgInvalids.1 = 0
cikeTunInP2ExchgRejects.1 = 0
cikeTunInP2SaDelRequests.1 = 0
cikeTunOutOctets.1 = 344
cikeTunOutPkts.1 = 2
cikeTunOutDropPkts.1 = 0
cikeTunOutNotifys.1 = 0
cikeTunOutP2Exchgs.1 = 1
cikeTunOutP2ExchgInvalids.1 = 0
cikeTunOutP2ExchgRejects.1 = 0
cikeTunOutP2SaDelRequests.1 = 0
cikeTunStatus.1 = active(1)
cikePeerCorrIpSecTunIndex.1.15.48.49.48.46.48.48.49.46.48.48.49.46.48.48.50.1.15.48.49.48.
46.48.48.49.46.48.48.49.46.48.48.49.1.1 = 1
cipSecGlobalActiveTunnels.0 = 1
cipSecGlobalPreviousTunnels.0 = 0
cipSecGlobalInOctets.0 = 400
cipSecGlobalHcInOctets.0 = 0x0190
cipSecGlobalInOctWraps.0 = 0
cipSecGlobalInDecompOctets.0 = 400
cipSecGlobalHcInDecompOctets.0 = 0x0190
cipSecGlobalInDecompOctWraps.0 = 0
cipSecGlobalInPkts.0 = 4
cipSecGlobalInDrops.0 = 0
cipSecGlobalInReplayDrops.0 = 0
cipSecGlobalInAuths.0 = 4
cipSecGlobalInAuthFails.0 = 0
cipSecGlobalInDecrypts.0 = 4
cipSecGlobalInDecryptFails.0 = 0
cipSecGlobalOutOctets.0 = 704
cipSecGlobalHcOutOctets.0 = 0x02c0
cipSecGlobalOutOctWraps.0 = 0
cipSecGlobalOutUncompOctets.0 = 704
cipSecGlobalHcOutUncompOctets.0 = 0x02c0
cipSecGlobalOutUncompOctWraps.0 = 0
cipSecGlobalOutPkts.0 = 4
cipSecGlobalOutDrops.0 = 0
cipSecGlobalOutAuths.0 = 4
cipSecGlobalOutAuthFails.0 = 0
cipSecGlobalOutEncrypts.0 = 4
cipSecGlobalOutEncryptFails.0 = 0
cipSecGlobalProtocolUseFails.0 = 0
cipSecGlobalNoSaFails.0 = 0
cipSecGlobalSysCapFails.0 = 0
cipSecTunIkeTunnelIndex.1 = 1
cipSecTunIkeTunnelAlive.1 = true(1)
cipSecTunLocalAddr.1 = 0a 01  01 02
cipSecTunRemoteAddr.1 = 0a 01  01 01
cipSecTunKeyType.1 = ike(1)
cipSecTunEncapMode.1 = tunnel(1)
cipSecTunLifeSize.1 = 99000
cipSecTunLifeTime.1 = 5000
cipSecTunActiveTime.1 = 13749
cipSecTunSaLifeSizeThreshold.1 = 64
cipSecTunSaLifeTimeThreshold.1 = 10
cipSecTunTotalRefreshes.1 = 0
cipSecTunExpiredSaInstances.1 = 0
cipSecTunCurrentSaInstances.1 = 4
cipSecTunInSaDiffHellmanGrp.1 = dhGroup1(2)
```

```
cipSecTunInSaEncryptAlgo.1 = des(2)
cipSecTunInSaAhAuthAlgo.1 = hmacSha(3)
cipSecTunInSaEspAuthAlgo.1 = hmacSha(3)
cipSecTunInSaDecompAlgo.1 = none(1)
cipSecTunOutSaDiffHellmanGrp.1 = dhGroup1(2)
cipSecTunOutSaEncryptAlgo.1 = des(2)
cipSecTunOutSaAhAuthAlgo.1 = hmacSha(3)
cipSecTunOutSaEspAuthAlgo.1 = hmacSha(3)
cipSecTunOutSaCompAlgo.1 = none(1)
cipSecTunInOctets.1 = 400
cipSecTunHcInOctets.1 = 0x0190
cipSecTunInOctWraps.1 = 0
cipSecTunInDecompOctets.1 = 400
cipSecTunHcInDecompOctets.1 = 0x0190
cipSecTunInDecompOctWraps.1 = 0
cipSecTunInPkts.1 = 4
cipSecTunInDropPkts.1 = 0
cipSecTunInReplayDropPkts.1 = 0
cipSecTunInAuths.1 = 4
cipSecTunInAuthFails.1 = 0
cipSecTunInDecrypts.1 = 4
cipSecTunInDecryptFails.1 = 0
cipSecTunOutOctets.1 = 704
cipSecTunHcOutOctets.1 = 0x02c0
cipSecTunOutOctWraps.1 = 0
cipSecTunOutUncompOctets.1 = 704
cipSecTunHcOutUncompOctets.1 = 0x02c0
cipSecTunOutUncompOctWraps.1 = 0
cipSecTunOutPkts.1 = 4
cipSecTunOutDropPkts.1 = 0
cipSecTunOutAuths.1 = 4
cipSecTunOutAuthFails.1 = 0
cipSecTunOutEncrypts.1 = 4
cipSecTunOutEncryptFails.1 = 0
cipSecTunStatus.1 = active(1)
cipSecEndPtLocalName.1.1 =
cipSecEndPtLocalType.1.1 = singleIpAddr(1)
cipSecEndPtLocalAddr1.1.1 = 16 01  01 01
cipSecEndPtLocalAddr2.1.1 = 16 01  01 01
cipSecEndPtLocalProtocol.1.1 = 0
cipSecEndPtLocalPort.1.1 = 0
cipSecEndPtRemoteName.1.1 =
cipSecEndPtRemoteType.1.1 = singleIpAddr(1)
cipSecEndPtRemoteAddr1.1.1 = 14 01  01 01
cipSecEndPtRemoteAddr2.1.1 = 14 01  01 01
cipSecEndPtRemoteProtocol.1.1 = 0
cipSecEndPtRemotePort.1.1 = 0
cipSecSpiDirection.1.1 = in(1)
cipSecSpiDirection.1.2 = out(2)
cipSecSpiDirection.1.3 = in(1)
cipSecSpiDirection.1.4 = out(2)
cipSecSpiValue.1.1 = 3891970674
cipSecSpiValue.1.2 = 1963217493
cipSecSpiValue.1.3 = 3691920464
cipSecSpiValue.1.4 = 3458912974
cipSecSpiProtocol.1.1 = ah(1)
cipSecSpiProtocol.1.2 = ah(1)
cipSecSpiProtocol.1.3 = esp(2)
cipSecSpiProtocol.1.4 = esp(2)
cipSecSpiStatus.1.1 = active(1)
cipSecSpiStatus.1.2 = active(1)
cipSecSpiStatus.1.3 = active(1)
cipSecSpiStatus.1.4 = active(1)
cipSecHistTableSize.0 = 200
```

```
cipSecHistCheckPoint.0 = ready(1)
cipSecFailTableSize.0 = 200
cipSecTrapCntlIkeTunnelStart.0 = enabled(1)
cipSecTrapCntlIkeTunnelStop.0 = enabled(1)
cipSecTrapCntlIkeSysFailure.0 = disabled(2)
cipSecTrapCntlIkeCertCrlFailure.0 = disabled(2)
cipSecTrapCntlIkeProtocolFail.0 = disabled(2)
cipSecTrapCntlIkeNoSa.0 = disabled(2)
cipSecTrapCntlIpSecTunnelStart.0 = enabled(1)
cipSecTrapCntlIpSecTunnelStop.0 = enabled(1)
cipSecTrapCntlIpSecSysFailure.0 = disabled(2)
cipSecTrapCntlIpSecSetUpFailure.0 = disabled(2)
cipSecTrapCntlIpSecEarlyTunTerm.0 = disabled(2)
cipSecTrapCntlIpSecProtocolFail.0 = disabled(2)
cipSecTrapCntlIpSecNoSa.0 = disabled(2)
orcas:14>
orcas:14>
orcas:14>
```

VRF abc2 Polled

Polling VRF abc2 results in the following output:

Note The ping was completed for VRF abc1 only. Therefore, the counters of VRF abc2 should remain in the initialized state.

```
setenv SR_UTIL_COMMUNITY abc2
orcas:15>
orcas:15> /auto/sw/packages/snmpr/10.14.2.0/solaris2bin/getmany -v2c 10.1.38.25
cipSecMIBObjects
cipSecMibLevel.0 = 1
cikeGlobalActiveTunnels.0 = 0
cikeGlobalPreviousTunnels.0 = 0
cikeGlobalInOctets.0 = 0
cikeGlobalInPkts.0 = 0
cikeGlobalInDropPkts.0 = 0
cikeGlobalInNotifys.0 = 0
cikeGlobalInP2Exchgs.0 = 0
cikeGlobalInP2ExchgInvalids.0 = 0
cikeGlobalInP2ExchgRejects.0 = 0
cikeGlobalInP2SaDelRequests.0 = 0
cikeGlobalOutOctets.0 = 0
cikeGlobalOutPkts.0 = 0
cikeGlobalOutDropPkts.0 = 0
cikeGlobalOutNotifys.0 = 0
cikeGlobalOutP2Exchgs.0 = 0
cikeGlobalOutP2ExchgInvalids.0 = 0
cikeGlobalOutP2ExchgRejects.0 = 0
cikeGlobalOutP2SaDelRequests.0 = 0
cikeGlobalInitTunnels.0 = 0
cikeGlobalInitTunnelFails.0 = 0
cikeGlobalRespTunnelFails.0 = 0
cikeGlobalSysCapFails.0 = 0
cikeGlobalAuthFails.0 = 0
cikeGlobalDecryptFails.0 = 0
cikeGlobalHashValidFails.0 = 0
cikeGlobalNoSaFails.0 = 0
cipSecGlobalActiveTunnels.0 = 0
cipSecGlobalPreviousTunnels.0 = 0
cipSecGlobalInOctets.0 = 0
cipSecGlobalHcInOctets.0 = 0x00
```

```
cipSecGlobalInOctWraps.0 = 0
cipSecGlobalInDecompOctets.0 = 0
cipSecGlobalHcInDecompOctets.0 = 0x00
cipSecGlobalInDecompOctWraps.0 = 0
cipSecGlobalInPkts.0 = 0
cipSecGlobalInDrops.0 = 0
cipSecGlobalInReplayDrops.0 = 0
cipSecGlobalInAuths.0 = 0
cipSecGlobalInAuthFails.0 = 0
cipSecGlobalInDecrypts.0 = 0
cipSecGlobalInDecryptFails.0 = 0
cipSecGlobalOutOctets.0 = 0
cipSecGlobalHcOutOctets.0 = 0x00
cipSecGlobalOutOctWraps.0 = 0
cipSecGlobalOutUncompOctets.0 = 0
cipSecGlobalHcOutUncompOctets.0 = 0x00
cipSecGlobalOutUncompOctWraps.0 = 0
cipSecGlobalOutPkts.0 = 0
cipSecGlobalOutDrops.0 = 0
cipSecGlobalOutAuths.0 = 0
cipSecGlobalOutAuthFails.0 = 0
cipSecGlobalOutEncrypts.0 = 0
cipSecGlobalOutEncryptFails.0 = 0
cipSecGlobalProtocolUseFails.0 = 0
cipSecGlobalNoSaFails.0 = 0
cipSecGlobalSysCapFails.0 = 0
cipSecHistTableSize.0 = 200
cipSecHistCheckPoint.0 = ready(1)
cipSecFailTableSize.0 = 200
cipSecTrapCntlIkeTunnelStart.0 = enabled(1)
cipSecTrapCntlIkeTunnelStop.0 = enabled(1)
cipSecTrapCntlIkeSysFailure.0 = disabled(2)
cipSecTrapCntlIkeCertCrlFailure.0 = disabled(2)
cipSecTrapCntlIkeProtocolFail.0 = disabled(2)
cipSecTrapCntlIkeNoSa.0 = disabled(2)
cipSecTrapCntlIpSecTunnelStart.0 = enabled(1)
cipSecTrapCntlIpSecTunnelStop.0 = enabled(1)
cipSecTrapCntlIpSecSysFailure.0 = disabled(2)
cipSecTrapCntlIpSecSetUpFailure.0 = disabled(2)
cipSecTrapCntlIpSecEarlyTunTerm.0 = disabled(2)
cipSecTrapCntlIpSecProtocolFail.0 = disabled(2)
cipSecTrapCntlIpSecNoSa.0 = disabled(2)
orcas:16>
```

CHAPTER 30

Troubleshooting the IPSec VPN SPA

This chapter describes techniques that you can use to troubleshoot the operation of your IPSec VPN SPAs in a Cisco 7600 series router.

It includes the following sections:

General Troubleshooting Information

This section describes general information for troubleshooting the IPSec VPN SPA and the Cisco 7600 SSC-400. It includes the following sections:

Interpreting Console Error Messages

Cisco 7600 series routers can generate error messages and other system messages to inform the operator of events that might require attention. These messages can be displayed on the console, or sent to a logging host using the System Logging (Syslog) protocol or Simple Network Management Protocol (SNMP).

System error messages are organized in the documentation according to the particular system facility that produces the messages. The IPSec VPN SPA and Cisco 7600 SSC-400 use the following facility names in error messages:

- IPSec VPN SPA—SPA_IPSEC
- Cisco 7600 SSC-400—C7600_SSC400

To view the explanations and recommended actions for Cisco 7600 series router error messages, including messages related to Cisco 7600 series router SIPs and SPAs, refer to the following documents:

- *Cisco 7600 Series Cisco IOS System Message Guide*
- *Cisco IOS System Error Messages*

Using debug Commands

Along with the other **debug** commands supported on the Cisco 7600 series router, you can obtain specific debug information for SIPs and SPAs on the Cisco 7600 series router using the **debug hw-module subslot** privileged EXEC command.

The **debug hw-module subslot** command is intended for use by Cisco Systems technical support personnel.

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support personnel. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

For more information about the crypto debugging commands available, see the "Using Crypto Conditional Debug" section on page 30-32.

For information about other **debug** commands supported on the Cisco 7600 series router, refer to the *Cisco IOS Debug Command Reference*.

Using show Commands

There are several **show** commands that you can use to monitor and troubleshoot the IPSec VPN SPA on the Cisco 7600 series router.

For more information about **show** commands to verify and monitor the IPSec VPN SPA, refer to the *Cisco IOS Software Releases 12.2SR Command Reference*s and to the *Cisco IOS Software Releases 12.2SX Command References*.

For more information about security-related **show** commands, refer to the *Cisco IOS Security Command Reference*.

Monitoring the IPSec VPN SPA

This section describes commands that can be used to display information about the IPSec VPN SPA hardware and configuration. It consists of the following subsections:

Displaying IPSec VPN SPA Hardware and System Information

Use the following commands to display hardware and system information:

- **show diagbus**—See the "Displaying Information About IPSec VPN SPA Ports" section on page 30-3.
- **show crypto engine accelerator statistic slot**—See the "Displaying Platform and Network Interface Controller Statistics for the IPSec VPN SPA" section on page 30-3.
- **show hw-module slot fpd**—See the "Displaying Information About Hardware Revision Levels" section on page 30-5.

Displaying Information About IPSec VPN SPA Ports

To display information about the type of SPAs that are installed in the router, use the **show diagbus** command.

The following example shows output from the **show diagbus** command on a Cisco 7600 series router with an IPSec VPN SPA installed in subslot 1 of a Cisco 7600 SSC-400 that is installed in slot 5:

```
Router# show diagbus

Slot 5: Logical_index 10
        2-subslot Services SPA Carrier-400 controller
        Board is analyzed ipc ready
        HW rev 0.3, board revision A01
        Serial Number: abc Part number: 73-6348-01

        Slot database information:
        Flags: 0x2004   Insertion time: 0x3DB5F4BC (4d20h ago)

        Controller Memory Size:
                248 MBytes CPU Memory
                8 MBytes Packet Memory
                256 MBytes Total on Board SDRAM
    Cisco IOS Software, cwlc Software (smsc-DWDBG-M), Version 12.2(nightly.SRA060615)
    NIGHTLY BUILD, synched to rainier RAINER_BASE

        SPA Information:
        subslot 5/1: SPA-IPSEC-2G (0x3D7), status: ok
```

Displaying Platform and Network Interface Controller Statistics for the IPSec VPN SPA

To display platform statistics, and, optionally, network interface controller statistics, use the **show crypto engine accelerator statistic slot** command.

The following example shows the platform statistics for the IPSec VPN SPA in slot 1, subslot 0, and also displays the network interface controller statistics:

```
Router# show crypto engine accelerator statistic slot 1/0 detail

VPN module in slot 1/0


Decryption Side Data Path Statistics
====================================
Packets RX...............: 454260
Packets TX...............: 452480
```

```
IPSec Transport Mode.....: 0
IPSec Tunnel Mode........: 452470
AH Packets...............: 0
ESP Packets..............: 452470
GRE Decapsulations.......: 0
NAT-T Decapsulations.....: 0
Clear....................: 8
ICMP.....................: 0

Packets Drop.............: 193
Authentication Errors....: 0
Decryption Errors........: 0
Replay Check Failed......: 0
Policy Check Failed......: 0
Illegal CLear Packet.....: 0
GRE Errors...............: 0
SPD Errors...............: 0
HA Standby Drop..........: 0

Hard Life Drop...........: 0
Invalid SA...............: 191
SPI No Match.............: 0
Destination No Match.....: 0
Protocol No Match........: 0

Reassembly Frag RX.......: 0
IPSec Fragments..........: 0
IPSec Reasm Done.........: 0
Clear Fragments..........: 0
Clear Reasm Done.........: 0
Datagrams Drop...........: 0
Fragments Drop...........: 0


Decryption Side Controller Statistics

=====================================
Frames RX................: 756088
Bytes RX.................: 63535848
Mcast/Bcast Frames RX....: 2341
RX Less 128Bytes.........: 756025
RX Less 512Bytes.........: 58
RX Less 1KBytes..........: 2
RX Less 9KBytes..........: 3
RX Frames Drop...........: 0

Frames TX................: 452365
Bytes TX.................: 38001544
Mcast/Bcast Frames TX....: 9
TX Less 128Bytes.........: 452343
TX Less 512Bytes.........: 22
TX Less 1KBytes..........: 0
TX Less 9KBytes..........: 0


Encryption Side Data Path Statistics
====================================
Packets RX...............: 756344
Packets TX...............: 753880
IPSec Transport Mode.....: 0
IPSec Tunnel Mode........: 753869
GRE Encapsulations.......: 0
NAT-T Encapsulations.....: 0
LAF prefragmented........: 0
```

```
Fragmented...............: 0
Clear....................: 753904
ICMP.....................: 0

Packets Drop.............: 123
IKE/TED Drop.............: 27
Authentication Errors....: 0
Encryption Errors........: 0
HA Standby Drop..........: 0

Hard Life Drop...........: 0
Invalid SA...............: 191

Reassembly Frag RX.......: 0
Clear Fragments..........: 0
Clear Reasm Done.........: 0
Datagrams Drop...........: 0
Fragments Drop...........: 0


Encryption Side Controller Statistics
=====================================
Frames RX................: 454065
Bytes RX.................: 6168274/
Mcast/Bcast Frames RX....: 1586
RX Less 128Bytes.........: 1562
RX Less 512Bytes.........: 452503
RX Less 1KBytes..........: 0
RX Less 9KBytes..........: 0
RX Frames Drop...........: 0

Frames TX................: 753558
Bytes TX.................: 100977246
Mcast/Bcast Frames TX....: 2
TX Less 128Bytes.........: 3
TX Less 512Bytes.........: 753555
TX Less 1KBytes..........: 0
TX Less 9KBytes..........: 0
```

Displaying Information About Hardware Revision Levels

To display information about the hardware revision of the Cisco 7600 SSC-400 and the IPSec VPN SPA as well as the version of the field-programmable devices (FPDs) that are on the carrier card and the SPA, use the **show hw-module slot fpd** command. Cisco technical engineers might need this information to debug or troubleshoot problems with a SPA installation.

The following example shows output from the **show hw-module slot fpd** command on a Cisco 7600 series router with an IPSec VPN SPA installed in subslot 0 of a Cisco 7600 SSC-400 that is installed in slot 6:

```
Router# show hw-module slot 6 fpd
==== ============ ==== ====================================================
     H/WField ProgrammableCurrentMin. Required
SlotCard TypeVer.Device: "ID-Name"VersionVersion
===== =========== ==== ===================== ================ ================
6   7600-SSC-4000.51-I/O FPGA1.01.0
---- ------------ ---- ------------------- ---------------- ---------------
6/0 SPA-IPSEC-2G0.31-PROM1.11.1
==== ============ ===== ======================================================
```

Displaying IPSec VPN SPA Configuration Information

Use the following commands to display information about the IPSec VPN SPA configuration:

- **show crypto vlan**—See the "Displaying Information About Access and Routed Ports That Are Connected" section on page 30-7, the "Displaying the VPN Running State" section on page 30-8, and the "Displaying Information About IP Multicast Over a GRE Tunnel" section on page 30-12.
- **show interfaces trunk**—See the "Displaying Information About the VLANs Allowed by a Trunk Port" section on page 30-7.
- **show ip route**—See the "Displaying the Routing Table" section on page 30-8.
- **show interfaces tunnel**—See the "Displaying Tunnel Interface Information" section on page 30-8.
- **show ip mroute**—See the "Displaying Information About IP Multicast Over a GRE Tunnel" section on page 30-12.
- **show crypto map**—See the "Displaying Information About Crypto Maps" section on page 30-13.
- **show crypto ipsec sa**—See the "Displaying Information About IPSec Security Associations" section on page 30-13.
- **show crypto isakmp sa**—See the "Displaying Information About SAs at a Peer" section on page 30-15.
- **show crypto session**—See the "Displaying Information About Crypto Sessions" section on page 30-15.
- **show crypto isakmp policy**—See the "Displaying Information About IKE Policies" section on page 30-16.
- **show crypto ipsec transform-set**—See the "Displaying Information About IPSec Transform Sets" section on page 30-16.
- **show call admission statistics**—See the "Displaying Call Admission Control (CAC) Information" section on page 30-17.
- **show crypto call admission statistics**—See the "Displaying Call Admission Control (CAC) Information" section on page 30-17.
- **show crypto key mypubkey rsa**—See the "Displaying Information About RSA Public Keys" section on page 30-17.
- **show crypto key pubkey-chain rsa**—See the "Displaying Information About RSA Public Keys" section on page 30-17.
- **show crypto pki trustpoints**—See the "Displaying Information About Trustpoints" section on page 30-18.
- **show crypto pki certificates storage**—See the "Displaying the Certificate Storage Location" section on page 30-18.
- **show crypto pki certificates**—See the "Displaying Information About Certificates" section on page 30-18.
- **show crypto pki server**—See the "Displaying Information About the Certificate Server" section on page 30-20.
- **show ip nhrp**—See the "Displaying Information About the NHRP Cache" section on page 30-20.
- **show crypto isakmp ha standby**—See the "Displaying HSRP Information" section on page 30-21.
- **show crypto ipsec ha**—See the "Displaying HSRP Information" section on page 30-21.
- **show crypto ipsec sa**—See the "Displaying HSRP Information" section on page 30-21.

- **show crypto ipsec sa standby**—See the "Displaying HSRP Information" section on page 30-21.
- **show ssp client**—See the "Displaying SSP Information" section on page 30-23.
- **show ssp packet**—See the "Displaying SSP Information" section on page 30-23.
- **show ssp peers**—See the "Displaying SSP Information" section on page 30-23.
- **show ssp redundancy**—See the "Displaying SSP Information" section on page 30-23.
- **show redundancy linecard-group**—See the "Displaying Information About a BFG Configuration" section on page 30-24.
- **show crypto ace redundancy**—See the "Displaying Information About a BFG Configuration" section on page 30-24.

For a detailed description of the information displayed by the **show** commands, refer to the "IP Security and Encryption" chapter of the *Cisco IOS Security Command Reference*.

Displaying Information About Access and Routed Ports That Are Connected

To verify that an access or routed port is connected, use the **show crypto vlan** command. The following is sample output from the command:

```
Router# show crypto vlan

Interface VLAN 100 on IPSec Service Module port Gi5/0/1 connected to VLAN 2022 with crypto
map set coral2


Router# show crypto vlan

Interface VLAN 100 on IPSec Service Module port Gi5/0/1 connected to Gi2/8 with crypto
mark set M10K
```

Displaying Information About the VLANs Allowed by a Trunk Port

To display information about the VLANs allowed by a trunk port, use the **show interfaces trunk** command. The following is sample output from the command:

```
Router# show interfaces GigabitEthernet 2/0/1 trunk

Port      Mode         Encapsulation  Status        Native vlan
Gi2/1     on           802.1q         trunking      1

Port      Vlans allowed on trunk
Gi2/1     1-4094

Port      Vlans allowed and active in management domain
Gi2/1     1-4,7-8,513,1002-1005

Port      Vlans in spanning tree forwarding state and not pruned
Gi2/1     1-4,7-8,513,1002-1005
```

Displaying the VPN Running State

To display the VPN running state, use the **show crypto vlan** command. The following is sample output from the command:

In the following example, the interface VLAN belongs to the IPSec VPN SPA inside port:

```
Router# show crypto vlan

  Interface VLAN 2 on IPSec Service Module port 7/0/1 connected to Fa8/3
```

In the following example, VLAN 2 is the interface VLAN and VLAN 2022 is the hidden VLAN:

```
Router# show crypto vlan

Interface VLAN 2 on IPSec Service Module port 3/0/1 connected to VLAN 2022 with crypto map
set coral2
```

In the following example, either the interface VLAN is missing on the IPSec VPN SPA inside port, the IPSec VPN SPA is removed from the chassis, or the IPSec VPN SPA was moved to a different subslot:

```
Router# show crypto vlan

  Interface VLAN 2 connected to VLAN 3 (no IPSec Service Module attached)
```

Displaying the Routing Table

To display the current state of the routing table, use the **show ip route** command. The following is sample output from the command:

```
Router1# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C 10.0.35.0/24 is directly connected, Ethernet3/3
S 10.0.36.0/24 is directly connected, Tunnel0
C 10.0.51.0/24 is directly connected, Tunnel0
C 10.0.149.0/24 is directly connected, Ethernet3/0
```

Displaying Tunnel Interface Information

To display tunnel interface information, use the **show interfaces tunnel** command. The following is sample output from the command:

```
Router# show interfaces tunnel 1

Tunnel4 is up, line protocol is down
Hardware is Routing Tunnel
Internet address is 10.1.1.1/24
MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec, rely 255/255, load 1/255
Encapsulation TUNNEL, loopback not set
Keepalive set (10 sec)
Tunnel source 9.2.2.1, destination 6.6.6.2
Tunnel protocol/transport GRE/IP, key disabled, sequencing disabled
```

```
Tunnel TOS 0xF, Tunnel TTL 128
Checksumming of packets disabled, fast tunneling enabled
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Queueing strategy, fifo
Output queue 0/0, 1 drops; input queue 0/75, 0 drops
30 second input rate 0 bits/sec, 0 packets/sec
30 second output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets, 0 restarts
```

Table 30-1 describes significant fields shown in the display.

Table 30-1 show interfaces tunnel Field Descriptions

Field	Description
Tunnel is {up | down}	Interface is currently active and inserted into ring (up) or inactive and not inserted (down).
line protocol is {up | down | administratively down}	Shows line protocol up if a valid route is available to the tunnel destination. Shows line protocol down if no route is available, or if the route would be recursive.
Hardware	Specifies the hardware type.
MTU	Maximum transmission unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100 percent reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method is always TUNNEL for tunnels.
loopback	Indicates whether loopback is set or not.
Keepalive	Indicates whether keepalives are set or not.
Tunnel source	IP address used as the source address for the tunnel packets.
destination	IP address of the tunnel destination.
Tunnel protocol	Tunnel transport protocol (the protocol the tunnel is using). This is based on the **tunnel mode** command, which defaults to GRE.
key	(Optional) ID key for the tunnel interface.
sequencing	(Optional) Indicates whether the tunnel interface drops datagrams that arrive out of order.

Table 30-1 show interfaces tunnel Field Descriptions (continued)

Field	Description
Last input	Number of hours, minutes, and seconds (or never) since the last packet was successfully received by an interface and processed locally on the router. Useful for knowing when a dead interface failed. This field is not updated by fast-switched traffic.
output	Number of hours, minutes, and seconds (or never) since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the “last” fields exceeds 24 hours, the number of days and hours is displayed. If that field overflows, asterisks are displayed.
Last clearing	Time at which the counters that measure cumulative statistics (such as number of bytes transmitted and received) shown in this report were last reset to zero. Note that variables that might affect routing (for example, load and reliability) are not cleared when the counters are cleared. Three asterisks (***) indicate the elapsed time is too large to be displayed. 0:00:00 indicates the counters were cleared more than 231 ms (and less than 232 ms) ago.
Output queue, drops Input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped because of a full queue.
30 second input rate, 30 second output rate	Average number of bits and packets transmitted per second in the last 30 seconds. The 30-second input and output rates should be used only as an approximation of traffic per second during a given 30-second period. These rates are exponentially weighted averages with a time constant of 30 seconds. A period of four time constants must pass before the average will be within two percent of the instantaneous rate of a uniform stream of traffic over that period.
packets input	Total number of error-free packets received by the system.
bytes	Total number of bytes, including data and MAC encapsulation, in the error-free packets received by the system.

Table 30-1 show interfaces tunnel Field Descriptions (continued)

Field	Description
no buffer	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernet networks and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the minimum packet size of the medium.
giants	Number of packets that are discarded because they exceed the maximum packet size of the medium.
CRC	Cyclic redundancy checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus itself. A high number of CRCs is usually the result of a station transmitting bad data.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver’s ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. Compare with the no buffer count. These buffers are different than the system buffers mentioned in the no buffer description. Broadcast storms and bursts of noise can cause the ignored count to be increased.
abort	Illegal sequence of one bits on a serial interface. This usually indicates a clocking problem between the serial interface and the data link equipment.
packets output	Total number of messages transmitted by the system.
bytes	Total number of bytes, including data and MAC encapsulation, transmitted by the system.

Table 30-1 show interfaces tunnel Field Descriptions (continued)

Field	Description
underruns	Number of times that the far-end transmitter has been running faster than the near-end router's receiver can handle. This may never be reported on some interfaces.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	Number of messages retransmitted because of an Ethernet collision. This usually is the result of an overextended LAN (Ethernet or transceiver cable too long, more than two repeaters between stations, or too many cascaded multiport transceivers). Some collisions are normal. However, if your collision rate climbs to around 4 or 5 percent, you should consider verifying that there is no faulty equipment on the segment and moving some existing stations to a new segment. A packet that collides is counted only once in output packets.
interface resets	Number of times an interface has been reset. The interface may be reset by the administrator or automatically when an internal error occurs.
restarts	Number of times that the controller was restarted because of errors.

Displaying Information About IP Multicast Over a GRE Tunnel

To display information about an IP multicast over a GRE tunnel configuration, enter the **show crypto vlan** and **show ip mroute** commands.

Enter the **show crypto vlan** command to check that the tunnel has been taken over by the IPSec VPN SPA. The following is sample output from the command:

```
Router(config)# show crypto vlan

Interface VLAN 100 on IPSec Service Module port Gi7/0/1 connected to Po1 with crypto map
set map_t3
Tunnel15 is accelerated via IPSec SM in subslot 7/0
```

Enter the **show ip mroute** command and look for the "H" flag to check that the IP multicast traffic is hardware-switched. The following is sample output from the command:

```
Router# show ip mroute 230.1.1.5

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
```

```
T - SPT-bit set, J - Join SPT, M - MSDP created entry,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report, Z - Multicast Tunnel
Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 230.1.1.5), 01:23:45/00:03:16, RP 15.15.1.1, flags: SJC
Incoming interface: Null, RPF nbr 0.0.0.0
Outgoing interface list:
Tunnel15, Forward/Sparse-Dense, 00:25:47/00:03:16
(120.1.0.3, 230.1.1.5), 01:23:46/00:03:25, flags: T
Incoming interface: GigabitEthernet8/1, RPF nbr 0.0.0.0, RPF-MFD
Outgoing interface list:
Tunnel15, Forward/Sparse-Dense, 00:25:47/00:03:16, H
```

Displaying Information About Crypto Maps

To display information about crypto map configurations, use the **show crypto map** command. The following is sample output from the command:

```
Router# show crypto map

Crypto Map: "router-alice" idb: Ethernet0 local address: 172.21.114.123
Crypto Map "router-alice" 10 ipsec-isakmp
Peer = 172.21.114.67
Extended IP access list 141
access-list 141 permit ip
source: addr = 172.21.114.123/0.0.0.0
dest:   addr = 172.21.114.67/0.0.0.0
Current peer: 172.21.114.67
Security-association lifetime: 4608000 kilobytes/120 seconds
PFS (Y/N): N
Transform sets={t1,}
```

Displaying Information About IPSec Security Associations

To display information about IPSec security associations, use the **show crypto ipsec sa** command. The following is sample output from the command:

```
Router# show crypto ipsec sa

interface: Ethernet0

    Crypto map tag: router-alice, local addr. 172.21.114.123
    local ident (addr/mask/prot/port): (172.21.114.123/255.255.255.255/0/0)
    remote ident (addr/mask/prot/port): (172.21.114.67/255.255.255.255/0/0)
    current_peer: 172.21.114.67
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 10, #pkts encrypt: 10, #pkts digest 10
    #pkts decaps: 10, #pkts decrypt: 10, #pkts verify 10
    #send errors 10, #recv errors 0

    local crypto endpt.: 172.21.114.123, remote crypto endpt.: 172.21.114.67
    path mtu 1500, media mtu 1500
    current outbound spi: 20890A6F

    inbound esp sas:
    spi: 0x257A1039(628756537)
```

```
        transform: esp-des esp-md5-hmac,
        in use settings ={Tunnel,}
        slot: 0, conn id: 26, crypto map: router-alice
        sa timing: remaining key lifetime (k/sec): (4607999/90)
        IV size: 8 bytes
        replay detection support: Y

    inbound ah sas:

    outbound esp sas:
    spi: 0x20890A6F(545852015)
        transform: esp-des esp-md5-hmac,
        in use settings ={Tunnel,}
        slot: 0, conn id: 27, crypto map: router-alice
        sa timing: remaining key lifetime (k/sec): (4607999/90)
        IV size: 8 bytes
        replay detection support: Y
     outbound ah sas:

interface: Tunnel0

    Crypto map tag: router-alice, local addr. 172.21.114.123
    local ident (addr/mask/prot/port): (172.21.114.123/255.255.255.255/0/0)
    remote ident (addr/mask/prot/port): (172.21.114.67/255.255.255.255/0/0)
    current_peer: 172.21.114.67
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 10, #pkts encrypt: 10, #pkts digest 10
    #pkts decaps: 10, #pkts decrypt: 10, #pkts verify 10
    #send errors 10, #recv errors 0
    local crypto endpt.: 172.21.114.123, remote crypto endpt.: 172.21.114.67
    path mtu 1500, media mtu 1500
    current outbound spi: 20890A6F

    inbound esp sas:
    spi: 0x257A1039(628756537)
        transform: esp-des esp-md5-hmac,
        in use settings ={Tunnel,}
        slot: 0, conn id: 26, crypto map: router-alice
        sa timing: remaining key lifetime (k/sec): (4607999/90)
        IV size: 8 bytes
        replay detection support: Y

    inbound ah sas:

    outbound esp sas:
    spi: 0x20890A6F(545852015)
        transform: esp-des esp-md5-hmac,
        in use settings ={Tunnel,}
        slot: 0, conn id: 27, crypto map: router-alice
        sa timing: remaining key lifetime (k/sec): (4607999/90)
        IV size: 8 bytes
        replay detection support: Y

    outbound ah sas:
```

Displaying Information About SAs at a Peer

To display information about all current IKE SAs at a peer, use the **show crypto isakmp sa** command. The following is sample output from the command:

```
Router# show crypto isakmp sa

f_vrf/i_vrf    dst              src            state        conn-id    slot
```

Displaying Information About Crypto Sessions

To display status information for active crypto sessions, use the **show crypto session** command. The output will include the following:

- Interface
- IKE peer description, if available
- IKE SAs that are associated with the peer by which the IPSec SAs are created
- IPSec SAs serving the flows of a session

The following is sample output from the command:

```
Router# show crypto session detail

Crypto session current status

Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, X - IKE Extended Authentication

Interface: Ethernet1/0
Session status: UP-NO-IKE
Peer: 10.2.80.179/500 fvrf: (none) ivrf: (none)
Desc: My-manual-keyed-peer
Phase1_id: 10.2.80.179
IPSEC FLOW: permit ip host 10.2.80.190 host 10.2.80.179
Active SAs: 4, origin: manual-keyed crypto map
Inbound: #pkts dec'ed 0 drop 0 life (KB/Sec) 0/0
Outbound: #pkts enc'ed 0 drop 0 life (KB/Sec) 0/0

Interface: Ethernet1/2
Session status: DOWN
Peer: 10.1.1.1/500 fvrf: (none) ivrf: (none)
Desc: SJC24-2-VPN-Gateway
Phase1_id: 10.1.1.1
IPSEC FLOW: permit ip host 10.2.2.3 host 10.2.2.2
Active SAs: 0, origin: crypto map
Inbound: #pkts dec'ed 0 drop 0 life (KB/Sec) 0/0
Outbound: #pkts enc'ed 0 drop 0 life (KB/Sec) 0/0
IPSEC FLOW: permit ip 10.2.0.0/255.255.0.0 10.4.0.0/255.255.0.0
Active SAs: 0, origin: crypto map
Inbound: #pkts dec'ed 0 drop 0 life (KB/Sec) 0/0
Outbound: #pkts enc'ed 0 drop 0 life (KB/Sec) 0/0

Interface: Serial2/0.17
Session status: UP-ACTIVE
Peer: 10.1.1.5/500 fvrf: (none) ivrf: (none)
Desc: (none)
Phase1_id: 10.1.1.5
IKE SA: local 10.1.1.5/500 remote 10.1.1.5/500 Active
Capabilities:(none) connid:1 lifetime:00:59:51
IPSEC FLOW: permit ip host 10.1.1.5 host 10.1.2.5
Active SAs: 2, origin: dynamic crypto map
```

```
Inbound: #pkts dec'ed 4 drop 0 life (KB/Sec) 20085/171
Outbound: #pkts enc'ed 4 drop 0 life (KB/Sec) 20086/171
```

Displaying Information About IKE Policies

To display information about IKE policies, use the **show crypto isakmp policy** command. The following is sample output from the command:

```
Router# show crypto isakmp policy 1

    encr 3des
    authentication pre-share
    group 2
    crypto isakmp key cisco address 192.168.3.1
```

Note If a user enters an IKE encryption method that the hardware does not support, a warning message will be displayed in the **show crypto isakmp policy** command output:

```
WARNING: encryption hardware does not support the configured encryption method for ISAKMP
policy value
```

Displaying Information About IPSec Transform Sets

To display information about transform set configurations, use the **show crypto ipsec transform-set** command. The following is sample output from the command:

```
Router# show crypto ipsec transform-set

Transform set combined-des-md5: {esp-des esp-md5-hmac}
will negotiate = {Tunnel,},
Transform set t1: {esp-des esp-md5-hmac}
will negotiate = {Tunnel,},
Transform set t100: {ah-sha-hmac}
will negotiate = {Transport,},
Transform set t2: {ah-sha-hmac}
will negotiate = {Tunnel,},
{esp-des}
will negotiate = {Tunnel,},
```

Note If a user enters an IPSec transform that the hardware (the IPSec peer) does not support, a warning message will be displayed in the **show crypto ipsec transform-set** command output:

```
WARNING: encryption hardware does not support transform.
```

Displaying Call Admission Control (CAC) Information

To display Call Admission Control configuration information, enter the **show call admission statistics** and the **show crypto call admission statistics** commands.

The **show call admission statistics** command monitors the global CAC configuration parameters and the behavior of CAC. The following is sample output from the command:

```
Router# show call admission statistics
Total Call admission charges: 0, limit 25
Total calls rejected 12, accepted 51
Load metric: charge 0, unscaled 0
```

The **show crypto call admission statistics** command monitors crypto CAC statistics. The following is sample output from the command

```
Router# show crypto call admission statistics
-----------------------------------------------------------
               Crypto Call Admission Control Statistics
-----------------------------------------------------------
System Resource Limit: 0    Max IKE SAs 0
Total IKE SA Count:    0    active:     0   negotiating: 0
Incoming IKE Requests: 0    accepted:   0   rejected:    0
Outgoing IKE Requests: 0    accepted:   0   rejected:    0
Rejected IKE Requests: 0    rsrc low:   0   SA limit:    0
```

Displaying Information About RSA Public Keys

To display information the RSA public keys configured for your router, use the **show crypto key mypubkey rsa** command. The following is sample output from the command:

```
Router# show crypto key mypubkey rsa

% Key pair was generated at: 06:07:50 UTC Jan 13 1996

Key name: myrouter.example.com

 Usage: Encryption Key

 Key Data:

  00302017 4A7D385B 1234EF29 335FC973 2DD50A37 C4F4B0FD 9DADE748 429618D5

  18242BA3 2EDFBDD3 4296142A DDF7D3D8 08407685 2F2190A0 0B43F1BD 9A8A26DB

  07953829 791FCDE9 A98420F0 6A82045B 90288A26 DBC64468 7789F76E EE21
```

To display a list of all the RSA public keys stored on your router (including the public keys of peers that have sent your router their certificates during peer authentication for IPSec), or to display details of a particular RSA public key stored on your router, use the **show crypto key pubkey-chain rsa** command. The following is sample output from the command:

```
Router# show crypto key pubkey-chain rsa

Codes: M - Manually Configured, C - Extracted from certificate

Code  Usage         IP-address     Name

M     Signature     10.0.0.1       myrouter.example.com

M     Encryption    10.0.0.1       myrouter.example.com
```

```
C      Signature    172.16.0.1     routerA.example.com

C      Encryption   172.16.0.1     routerA.example.com

C      General      192.168.10.3   routerB.domain1.com
```

Displaying Information About Trustpoints

To display the trustpoints that are configured in the router, use the **show crypto pki trustpoints** command. The following is sample output from the command:

```
Router# show crypto pki trustpoints

Trustpoint bo:

    Subject Name:

    CN = bomborra Certificate Manager

     O = cisco.com

     C = US

          Serial Number:01

    Certificate configured.

    CEP URL:http://bomborra

    CRL query url:ldap://bomborra
```

Displaying the Certificate Storage Location

To display the current setting for the PKI certificate storage location, use the **show crypto pki certificates storage** command.

The following is sample output from the command:

```
Router# show crypto pki certificates storage

Certificates will be stored in disk0:/certs/
```

Displaying Information About Certificates

To display information about your certificate, the certificate of the CA, and any RA certificates, use the **show crypto pki certificates** command. The following is sample output from the command:

```
Router# show crypto pki certificates

CA Certificate

Status: Available
    Certificate Serial Number: 1244325DE0369880465F977A18F61CA8
    Certificate Usage: Signature

    Issuer:
```

```
        CN = new-user
        OU = pki new-user
        O = cisco
        L = santa cruz2
        ST = CA
        C = US
        EA = user@cysco.net

    Subject:
        CN = new-user
        OU = pki new-user
        O = cisco
        L = santa cruz2
        ST = CA
        C = US
        EA = user@cysco.net

    CRL Distribution Point:
        http://new-user.cysco.net/CertEnroll/new-user.crl

    Validity Date:
        start date: 14:19:29 PST Oct 31 2002
        end date: 14:27:27 PST Oct 31 2017

    Associated Trustpoints: MS

Certificate

Status: Available
    Certificate Serial Number: 193E28D20000000009F7
    Certificate Usage: Signature

    Issuer:
        CN = new-user
        OU = pki new-user
        O = cisco
        L = santa cruz2
        ST = CA
        C = US
        EA = user@cysco.net

    Subject:
        Name: User1.Cysco.Net

    CRL Distribution Point:
        http://new-user.cysco.net/CertEnroll/new-user.crl

    Validity Date:
        start date: 12:40:14 PST Feb 26 2003
        end   date: 12:50:14 PST Mar 5 2003
        renew date: 16:00:00 PST Dec 31 1969

    Associated Trustpoints: MS
```

Displaying Information About the Certificate Server

To display the current state and configuration of the certificate server, use the **show crypto pki server** command. The following is sample output from the command:

```
Router# show crypto pki server

Certificate Server status: disabled, storage configuration incomplete

    Granting mode is: manual

    Last certificate issued serial number: 0

    CA certificate expiration timer: 21:29:38 GMT Jun 5 2006

    CRL NextUpdate timer: 21:31:39 GMT Jun 6 2003

    Current storage dir: ftp://myftpserver

    Database Level: Minimum - no cert data written to storage
```

Displaying Information About the NHRP Cache

To display information about the Next Hop Resolution Protocol (NHRP) cache, use the **show ip nhrp** command. The following is sample output from the command:

```
Router# show ip nhrp

10.10.1.75/32 via 10.10.1.75, Tunnel5 created 00:32:11, expire 00:01:46

  Type: dynamic, Flags: authoritative unique registered

  NBMA address: 172.16.175.75

10.10.1.76/32 via 10.10.1.76, Tunnel5 created 00:26:41, expire 00:01:37

  Type: dynamic, Flags: authoritative unique registered

  NBMA address: 172.16.175.76

10.10.1.77/32 via 10.10.1.77, Tunnel5 created 00:31:26, expire 00:01:33

  Type: dynamic, Flags: authoritative unique registered

  NBMA address: 172.17.63.20
```

Displaying HSRP Information

To display information about HSRP configurations, use the **show crypto isakmp ha standby**, **show crypto ipsec ha**, **show ipsec sa**, and **show crypto ipsec sa standby** commands.

Enter the **show crypto isakmp ha standby** command to view your ISAKMP standby or active SAs. The following is sample output from the command:

```
Router# show crypto isakmp ha standby

dst             src             state       I-Cookie           R-Cookie

172.16.31.100   20.3.113.1      QM_IDLE     796885F3 62C3295E  FFAFBACD

EED41AFF

172.16.31.100   20.2.148.1      QM_IDLE     5B78D70F 3D80ED01  FFA03C6D

09FC50BE

172.16.31.100   20.4.124.1      QM_IDLE     B077D0A1 0C8EB3A0  FF5B152C

D233A1E0

172.16.31.100   20.3.88.1       QM_IDLE     55A9F85E 48CC14DE  FF20F9AE

DE37B913

172.16.31.100   20.1.95.1       QM_IDLE     3881DE75 3CF384AE  FF192CAB
```

Enter the **show crypto ipsec ha** command to view your IPSec High Availability Manager state. The following is sample output from the command:

```
Router# show crypto ipsec ha

Interface           VIP                SAs   IPSec HA State

FastEthernet0/0     172.16.31.100     1800   Active since 13:00:16 EDT Tue Oct 1 2002
```

Enter the **show crypto ipsec sa** command to view HA status of the IPSec SA (standby or active). The following is sample output from the command:

```
Router# show crypto ipsec sa

interface: FastEthernet0/0

    Crypto map tag: mymap, local addr. 172.168.3.100
    local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
    remote ident (addr/mask/prot/port): (5.6.0.0/255.255.0.0/0/0)
    current_peer: 172.168.3.1
    PERMIT, flags={}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

    local crypto endpt.: 172.168.3.100, remote crypto endpt.: 172.168.3.1
    path mtu 1500, media mtu 1500
    current outbound spi: 132ED6AB

    inbound esp sas:
    spi: 0xD8C8635F(3637011295)
        transform: esp-des esp-md5-hmac ,
```

```
        in use settings ={Tunnel, }
        slot: 0, conn id: 2006, flow_id: 3, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4499/59957)
        IV size: 8 bytes
        replay detection support: Y
        HA Status: STANDBY

    inbound ah sas:
    spi: 0xAAF10A60(2867923552)
        transform: ah-sha-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2004, flow_id: 3, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4499/59957)
        replay detection support: Y
        HA Status: STANDBY

    inbound pcp sas:

    outbound esp sas:
    spi: 0x132ED6AB(321836715)
        transform: esp-des esp-md5-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2007, flow_id: 4, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4499/59957)
        IV size: 8 bytes
        replay detection support: Y
        HA Status: STANDBY

    outbound ah sas:
    spi: 0x1951D78(26549624)
        transform: ah-sha-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2005, flow_id: 4, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4499/59957)
        replay detection support: Y
        HA Status: STANDBY

    outbound pcp sas:
```

Enter the **show crypto ipsec sa standby** command to view your standby SAs. The following is sample output from the command:

```
Router# show crypto ipsec sa standby

interface: FastEthernet0/0
    Crypto map tag: mymap, local addr. 172.168.3.100
    local  ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
    remote ident (addr/mask/prot/port): (5.6.0.0/255.255.0.0/0/0)
    current_peer: 172.168.3.1
    PERMIT, flags={}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

    local crypto endpt.: 172.168.3.100, remote crypto endpt.: 172.168.3.1
    path mtu 1500, media mtu 1500
    current outbound spi: 132ED6AB

    inbound esp sas:
    spi: 0xD8C8635F(3637011295)
        transform: esp-des esp-md5-hmac ,
        in use settings ={Tunnel, }
```

```
        slot: 0, conn id: 2006, flow_id: 3, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4499/59957)
        IV size: 8 bytes
        replay detection support: Y
        HA Status: STANDBY

    inbound ah sas:
    spi: 0xAAF10A60(2867923552)
        transform: ah-sha-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2004, flow_id: 3, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4499/59957)
        replay detection support: Y
        HA Status: STANDBY

    inbound pcp sas:

    outbound esp sas:
    spi: 0x132ED6AB(321836715)
        transform: esp-des esp-md5-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2007, flow_id: 4, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4499/59957)
        IV size: 8 bytes
        replay detection support: Y
        HA Status: STANDBY

    outbound ah sas:
    spi: 0x1951D78(26549624)
        transform: ah-sha-hmac ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 2005, flow_id: 4, crypto map: mymap
        sa timing: remaining key lifetime (k/sec): (4499/59957)
        replay detection support: Y
        HA Status: STANDBY

    outbound pcp sas:
```

Displaying SSP Information

To display information about an SSP configuration, use the **show ssp client**, **show ssp packet**, **show ssp peers**, and **show ssp redundancy** commands.

Enter the **show ssp client** command to display the domain of interpretation (DOI), name, running version and available version of each client that is registered with SSP. The following is sample output from the command:

```
Router# show ssp client

SSP Client Information

    DOI   Client Name                      Version   Running Ver

      1   IPSec HA Manager                  1.0       1.0

      2   IKE HA Manager                    1.0       1.0
```

Enter the **show ssp packet** command to display the byte count and packet count for the current socket, the creation time of the socket, the server port number, and the port number used for SSP communication. The following is sample output from the command:

```
Router# show ssp packet

SSP packet Information

    Socket creation time: 01:01:06

    Local port: 3249      Server port: 3249

    Packets Sent = 38559, Bytes Sent = 2285020

    Packets Received = 910, Bytes Received = 61472
```

Enter the **show ssp peers** command to display the IP address of the remote peer, the interface used, and the connection state. The following is sample output from the command:

```
Router# show ssp peers

SSP Peer Information

    IP Address      Connection State   Local Interface

    40.0.0.1        Connected          FastEthernet0/1
```

Enter the **show ssp redundancy** command to display the current SSP state, the HSRP group name, interface used, and the elapsed time since last state change. The following is sample output from the command:

```
Router# show ssp redundancy

SSP Redundancy Information

  Device has been ACTIVE for 02:55:34

    Virtual IP      Redundancy Name             Interface

    172.16.31.100   KNIGHTSOFNI                 FastEthernet0/0
```

Displaying Information About a BFG Configuration

To display information about a BFG configuration, use the **show redundancy linecard-group** and **show crypto ace redundancy** commands. The following is sample output from the commands:

```
Router# show redundancy linecard-group 1

Line Card Redundancy Group:1 Mode:feature-card
Class:load-sharing
Cards:
Slot:3 Subslot:0
Slot:5 Subslot:0
```

```
Router# show crypto ace redundancy
--------------------------------------
LC Redundancy Group ID             :1
Pending Configuration Transactions:0
Current State                      :OPERATIONAL
Number of blades in the group      :2
Slots
--------------------------------------
Slot:3 Subslot:0
Slot state:0x36
Booted
Received partner config
Completed Bulk Synchronization
Crypto Engine in Service
Rebooted 22 times
Initialization Timer not running
Slot:5 Subslot:0
Slot state:0x36
Booted
Received partner config
Completed Bulk Synchronization
Crypto Engine in Service
Rebooted 24 times
Initialization Timer not running

ACE B2B Group State:OPERATIONAL Event:BULK DONE
ACE B2B Group State:CREATED Event:CONFIG_DOWNLOAD_DONE
ACE B2B Group State:DELETED Event:CONFIG_DELETE
ACE B2B Group State:OPERATIONAL Event:BULK DONE
ACE B2B Group State:CREATED Event:CONFIG_DOWNLOAD_DONE
ACE B2B Group State:DELETED Event:CONFIG_DELETE
ACE B2B Group State:OPERATIONAL Event:CONFIG_DOWNLOAD_DONE
ACE B2B Group State:DELETED Event:CONFIG_ADD
ACE B2B Group State:CREATED Event:UNDEFINED B2B HA EVENT
ACE B2B Group State:CREATED Event:CONFIG_DOWNLOAD_DONE
```

Troubleshooting Specific Problems on the IPSec VPN SPA

This section provides additional information about troubleshooting specific problems related to the IPSec VPN SPA. It includes the following subsections:

Troubleshooting Trunk Port Configurations

Caution When you configure an Ethernet port as a trunk port, all the VLANs are allowed on the trunk port by default. This default configuration does not work well with the IPSec VPN SPA and causes network loops.

Troubleshooting VRF-Aware IPSec

To troubleshoot VRF-Aware IPSec, use the **debug crypto ipsec** and the **debug crypto isakmp** commands. The **debug crypto ipsec** command displays IP security events. The **debug crypto isakmp** command displays messages about IKE events.

Debug Examples for VRF-Aware IPSec

The following sample debug outputs are for a VRF-aware IPSec configuration:

```
Router# debug crypto ipsec

Crypto IPSEC debugging is on
IPSEC-PE# debug crypto isakmp
Crypto ISAKMP debugging is on
IPSEC-PE# debug crypto isakmp d
04:31:28: ISAKMP (0:12): purging SA., sa=6482B354, delme=6482B354
04:31:28: ISAKMP: Unlocking IKE struct 0x63C142F8 for declare_sa_dead(), count 0
IPSEC-PE# debug crypto isakmp detail
Crypto ISAKMP internals debugging is on
IPSEC-PE#
IPSEC-PE#
IPSEC-PE#
04:32:07: ISAKMP: Deleting peer node by peer_reap for 10.1.1.1: 63C142F8
04:32:55: ISAKMP cookie gen for src 172.16.1.1 dst 10.1.1.1
04:32:55: ISAKMP cookie 3123100B DC887D4E
04:32:55: ISAKMP cookie gen for src 10.1.1.1 dst 172.68.1.1
04:32:55: ISAKMP cookie AA8F7B41 49A60E88
04:32:55: ISAKMP cookie gen for src 172.16.1.1 dst 10.1.1.1
04:32:55: ISAKMP cookie 3123100B DBC8E125
04:32:55: ISAKMP cookie gen for src 10.1.1.1 dst 172.16.1.1
04:32:55: ISAKMP cookie AA8F7B41 B4BDB5B7
04:32:55: ISAKMP (0:0): received packet from 10.1.1.1 dport 500 sport 500 Global (N) NEW
SA
04:32:55: ISAKMP: local port 500, remote port 500
04:32:55: ISAKMP: hash from 729FA94 for 619 bytes
04:32:55: ISAKMP: Packet hash:
64218CC0:                   B91E2C70 095A1346          9.,p.Z.F
```

```
64218CD0: 0EDB4CA6 8A46784F B314FD3B 00        .[L&.FxO.};.
04:32:55: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:32:55: ISAKMP cookie AA8F7B41 F7ACF384
04:32:55: ISAKMP cookie gen for src 10.1.1.1 dst 172.18.1.1
04:32:55: ISAKMP cookie AA8F7B41 0C07C670
04:32:55: ISAKMP: insert sa successfully sa = 6482B354
04:32:55: ISAKMP (0:13): processing SA payload. message ID = 0
04:32:55: ISAKMP (0:13): processing ID payload. message ID = 0
04:32:55: ISAKMP (0:13): peer matches vpn2-ra profile
04:32:55: ISAKMP: Looking for a matching key for 10.1.1.1 in default
04:32:55: ISAKMP: Created a peer struct for 10.1.1.1, peer port 500
04:32:55: ISAKMP: Locking peer struct 0x640BBB18, IKE refcount 1 for
crypto_ikmp_config_initialize_sa
04:32:55: ISAKMP (0:13): Setting client config settings 648252B0
04:32:55: ISAKMP (0:13): (Re)Setting client xauth list  and state
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID seems Unity/DPD but major 157 mismatch
04:32:55: ISAKMP (0:13): vendor ID is NAT-T v3
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID seems Unity/DPD but major 123 mismatch
04:32:55: ISAKMP (0:13): vendor ID is NAT-T v2
04:32:55: ISAKMP (0:13) Authentication by xauth preshared
04:32:55: ISAKMP (0:13): Checking ISAKMP transform 1 against priority 1 policy
04:32:55: ISAKMP:      encryption 3DES-CBC
04:32:55: ISAKMP:      hash SHA
04:32:55: ISAKMP:      default group 2
04:32:55: ISAKMP:      auth XAUTHInitPreShared
04:32:55: ISAKMP:      life type in seconds
04:32:55: ISAKMP:      life duration (VPI) of  0x0 0x20 0xC4 0x9B
04:32:55: ISAKMP (0:13): atts are acceptable. Next payload is 3
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID seems Unity/DPD but major 157 mismatch
04:32:55: ISAKMP (0:13): vendor ID is NAT-T v3
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID seems Unity/DPD but major 123 mismatch
04:32:55: ISAKMP (0:13): vendor ID is NAT-T v2
04:32:55: ISAKMP (0:13): processing KE payload. message ID = 0
04:32:55: ISAKMP (0:13): processing NONCE payload. message ID = 0
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID is DPD
04:32:55: ISAKMP (0:13): processing vendor id payload
04:32:55: ISAKMP (0:13): vendor ID seems Unity/DPD but major 175 mismatch
04:32:55: ISAKMP (0:13): vendor ID is XAUTH
```

Troubleshooting GRE Tunneling

If you see a line protocol down, as in the following example, it might be because of a recursive route:

```
%TUN-RECURDOWN Interface Tunnel 0 temporarily disabled due to recursive routing
```

To avoid recursive routing problems, keep passenger and transport network routing information disjointed by using the following techniques:

- Using a different AS number or tag.
- Using a different routing protocol.
- Using static routes to override the first hop (but watch for routing loops).

Clearing (and Reinitializing) IPSec Security Associations

You can clear (and reinitialize) IPSec security associations by using the **clear crypto sa** command.

Using the **clear crypto sa** command without parameters will clear out the full SA database, which will clear out active security sessions. You may also specify the **peer**, **map**, or **entry** keywords to clear out only a subset of the SA database. For more information, refer to the **clear crypto sa** command in the *Cisco IOS Security Command Reference*.

Troubleshooting IKE Policy and Transform Sets

Any IPSec transforms or IKE encryption methods that the current hardware does not support should be disabled; they are ignored whenever an attempt to negotiate with the peer is made.

If a user enters an IPSec transform or an IKE encryption method that the hardware does not support, a warning message will be generated. These warning messages are also generated at boot time. When an encrypted card is inserted, the current configuration is scanned. If any IPSec transforms or IKE encryption methods are found that are not supported by the hardware, a warning message will be generated.

Troubleshooting ISAKMP Keyrings and Peer Filtering

If an ISAKMP profile or ISAKMP keyring fails to be selected, you should check the local-address binding in the ISAKMP profile or ISAKMP keyring configuration and follow the output of the IKE debugs to determine whether the peer is correctly terminating on the address. You may remove the local-address binding (to make the scope of the profile or keyring global) and check to determine whether the profile or keyring is selected to confirm the situation.

You can use the **debug crypto ipsec** and **debug crypto isakmp** commands to check your configurations.

Troubleshooting Certificate to ISAKMP Profile Mapping

To monitor and maintain certificate to ISAKMP profile mapping, use the **debug crypto isakmp** command in privileged EXEC mode:

Router# **debug crypto isakmp**

The **debug crypto isakmp** command displays output showing that the certificate has gone through certificate map matching and that the certificate matches the ISAKMP profile.

Note The **debug crypto isakmp** command may also be used to verify that the peer has been assigned a group.

Troubleshooting IKE Aggressive Mode Initiation

To troubleshoot the IKE Aggressive Mode Initiation, use the following debug commands in privileged EXEC mode:

Router# **debug aaa authorization**

The **debug aaa authorization** command displays information on AAA authorization.

Router# **debug crypto isakmp**

The **debug crypto isakmp** command displays messages about IKE events.

Router# **debug radius**

The **debug radius** command displays information associated with the RADIUS host.

Troubleshooting RRI

To observe the behavior of RRI and its relationship to the creation and deletion of an IPSec SA, use the **debug crypto ipsec** command.

Troubleshooting IPSec Anti-Replay Window Size

If your anti-replay window size has not been set to a number that is high enough for the number of packets received, you will receive a system message such as the following:

```
*Nov 17 19:27:32.279: %CRYPTO-4-PKT_REPLAY_ERR: decrypt: replay check failed

connection id=1
```

The above message is generated when a received packet is judged to be outside the anti-replay window.

Troubleshooting a Crypto Map-Based Distinguished Name Configuration

If an encrypting peer attempts to establish a connection that is blocked by the DN-based crypto map configuration, the following error message will be logged:

```
time: %CRYPTO-4-IKE_QUICKMODE_BAD_CERT: encrypted connection attempted with a
peer without the configured certificate attributes
```

Troubleshooting PKI AAA Authorization

If your PKI AAA authorization configuration is not functioning correctly, you should ensure that the AAA username being used by the router is the same as the username on the AAA server. Use the **debug crypto pki transactions** command to see which username is being used by the router, as shown in the following example:

```
Router# debug crypto pki transactions

Jul  9 18:11:28.462: CRYPTO_PKI: Found a issuer match
Jul  9 18:11:28.658: CRYPTO_PKI: Certificate validated
Jul  9 18:11:28.686: CRYPTO_PKI_AAA: checking AAA authorization (tac-e,
cn=jack,ou=PKI,o=Cisco Systems,c=US, <all>)
Jul  9 18:11:29.126: CRYPTO_PKI_AAA: authorization declined by AAA, or AAA server not
found.
Jul  9 18:11:29.126: CRYPTO_PKI_AAA: No cert-application attribute found. Failing.
Jul  9 18:11:29.126: CRYPTO_PKI_AAA: authorization failed
Jul  9 18:11:29.126: CRYPTO_PKI: AAA authorization for list 'tac-e', and user
'cn=jack,ou=PKI,o=Cisco Systems,c=US' failed.
Jul  9 18:11:29.126: CRYPTO_PKI: chain cert was anchored to trustpoint root, and chain
validation result was: CRYPTO_INVALID_CERT
```

Troubleshooting Source Interface Selection

If your source interface selection configuration is not functioning correctly, check the following:

- Ensure that the interface specified in the command has a valid address. Attempt to ping the router using the address of the specified interface from another device (possibly the HTTP or LDAP server that is serving the CRL). You can do the same thing by using a trace route to the router from the external device.
- You can also test connectivity between the router and the CA or LDAP server by using the Cisco IOS command-line interface (CLI). Enter the **ping ip** command and respond to the prompts. If you answer "yes" to the "**Extended commands [n]**:" prompt, you will be able to specify the source address or interface.
- In addition, you can use the Cisco IOS CLI to input a **traceroute** command. If you enter the **traceroute ip** command (in user or privileged EXEC mode), you will be prompted for the destination and source address. You should specify the CA or LDAP server as the destination and the address of the interface that you specified in the "source interface" as the source address.

Troubleshooting Easy VPN Remote RSA Signature

To troubleshoot your Easy VPN Remote RSA signature configuration, use the following debug commands. The debug commands can be used in any order or individually.

Router# **debug crypto ipsec client ezvpn**

The **debug crypto ipsec client ezvpn** command displays information about the VPN tunnel as it relates to the Easy VPN remote configuration.

Router# **debug crypto isakmp**

The **debug crypto isakmp** command displays messages about IKE events.

Troubleshooting IPSec Stateful Failover Using HSRP and SSP (VPN High Availability)

If you find that either the active or standby IPSec Stateful Failover (VPN High Availability) processes are dysfunctional, you can perform the following checks:

- Use the **show ssp** command to verify the SSP process is running.
- Make sure that both routers share identical IPSec configurations. This is critical. If routers are configured differently, IPSec Stateful Failover (VPN High Availability) will not work.
- Verify that an IPSec connection can be formed with existing maps, transforms, and access lists.
- Configure HSRP on the inside and outside interfaces and make the HSRP groups track one another. Verify this works properly by performing a **shut** command on either of the interfaces, then observe that the HSRP standby router takes active control from the active router.
- Verify that SSP peers can see each other by performing a **show ssp peer** command on both the active and standby routers.
- Bind the IKE and IPSec to SSP and send traffic over the tunnels. A user can view High Availability (HA) messages on the standby router as both the active and standby routers synchronize.
- HSRP settings may require adjustments depending on the interface employed, such as Fast Ethernet or Gigabit Ethernet.

Checking HSRP Settings

To check HSRP settings, perform the following steps:

	Command	Purpose
Step 1	Router# **show standby brief**	Ensures that the interfaces are synchronized.
Step 2	Router# **no standby delay timer**	Leaves the delay timers at their default settings
Step 3	Router# **show standby brief**	When the other router comes online, issue the **show standby brief** command once again. If the output shows an interface on standby, you must set the standby router's delay timer.

Clearing Dormant SAs on Standby Routers

To clear associated SA entries, perform the following commands:

	Command	Purpose
Step 1	Router# **clear crypto isakmp ha** [**standby**][**resync**]	Clears all dormant (standby) entries from the device. If the **resync** keyword is used, all standby IKE SAs will be removed, and a resynchronization of state will occur.
Step 2	Router# **clear crypto sa ha standby** [**peer** *ip address* | **resync**]	Clears all standby SAs for the device if **peer** is specified.

Enabling Debugging for HA

To enable debugging for HA, perform the following commands:

	Command	Purpose
Step 1	Router# **debug crypto isakmp ha** [**detail** | **fsm** | **update**]	Enables basic debug messages related to the IKE HA Manager.
Step 2	Router# **debug crypto ipsec ha** [**detail** | **fsm** | **update**]	Enables IPSec HA debugging.
Step 3	Router# **debug ssp** [**fsm** | **socket** | **packet** | **peers** | **redundancy** | **config**]	Enables SSP debugging.

Enabling the Buffer Log

To prevent debug messages from flooding the console, disable the console log and enable the buffer log as follows:

	Command	Purpose
Step 1	Router# **configure terminal**	Enters global configuration mode.
Step 2	Router(config)# **logging buffered**	Enables the buffer log.
Step 3	Router(config)# **no logging console**	Disables the console log.

Troubleshooting a Blade Failure Group

To enable IPSec VPN SPA debugging for a Blade Failure Group, enter the **debug crypto ace b2b** command:

```
Router# debug crypto ace b2b

ACE B2B Failover debugging is on
```

Troubleshooting IPSec and IKE MIB Support for Cisco VRF-Aware IPSec Feature

The following **debug crypto mib** command and keywords may be used to display information about the IPsec and Internet Key Exchange (IKE) MIB as it relates to Cisco VRF-aware IPSec:

Router# **debug crypto mib** {**detail | error**}

In this command, the **detail** keyword displays different events as they occur in the IPsec MIB subsystem, whereas the **error** keyword displays error events in the MIB agent.

Note Due consideration should be given to enabling **debug crypto mib detail** because the output for the **detail** keyword can be quite long.

Using Crypto Conditional Debug

The Crypto Conditional Debug feature provides three command-line interfaces (CLIs) that allow you to debug an IP Security (IPSec) tunnel on the basis of predefined crypto conditions such as the peer IP address, connection-ID of a crypto engine, and security parameter index (SPI). By limiting debug messages to specific IPSec operations and reducing the amount of debug output, you can better troubleshoot a router with a large number of tunnels

The crypto conditional debug commands—**debug crypto condition**, **debug crypto condition unmatched**, and **show crypto debug-condition**—allow you to specify conditions (filter values) in which to generate and display debug messages related only to the specified conditions.

Table 30-2 lists the supported condition types.

Table 30-2 Supported Condition Types for Crypto Conditional Debug Commands

Condition Type (Keyword)	Description
connid	An integer between 1 and 32766. Relevant debug messages will be shown if the current IPSec operation uses this value as the connection-ID to interface with the crypto engine.
flowid	An integer between 1 and 32766. Relevant debug messages will be shown if the current IPSec operation uses this value as the flow-ID to interface with the crypto engine.

Table 30-2 Supported Condition Types for Crypto Conditional Debug Commands (continued)

Condition Type (Keyword)	Description
fvrf	The name string of a virtual private network (VPN) routing and forwarding (VRF) instance. Relevant debug messages will be shown if the current IPSec operation uses this VRF instance as its front door VRF (FVRF).
ivrf	The name string of a VRF instance. Relevant debug messages will be shown if the current IPSec operation uses this VRF instance as its inside VRF (IVRF).
peer group	A Unity group name string. Relevant debug messages will be shown if the peer is using this group name as its identity.
peer hostname	A fully qualified domain name (FQDN) string. Relevant debug messages will be shown if the peer is using this string as its identity.
peer ipv4	A single IP address. Relevant debug messages will be shown if the current IPSec operation is related to the IP address of this peer.
peer subnet	A subnet and a subnet mask that specify a range of peer IP addresses. Relevant debug messages will be shown if the IP address of the current IPSec peer falls into the specified subnet range.
peer username	A username string. Relevant debug messages will be shown if the peer is using this username as its identity.
spi	A 32-bit unsigned integer. Relevant debug messages will be shown if the current IPSec operation uses this value as the SPI.

Note If **connid**, **flowid**, or **spi** is used as a debug condition, the debug messages for a related IPSec flow are generated. An IPSec flow has two connection-IDs, flow-IDs, and SPI values—one inbound and one outbound. Either one of the two connection-IDs, flow-IDs, and SPI values can be used as the debug condition that triggers debug messages for the IPSec flow.

Crypto Conditional Debug Configuration Guidelines and Restrictions

Follow these guidelines and restrictions when configuring Crypto Conditional Debug:

- This feature does not support debug message filtering for hardware crypto engines.
- Although conditional debugging is useful for troubleshooting peer-specific or functionality-related Internet Key Exchange (IKE) and IPSec problems, conditional debugging may not be able to define and check large numbers of debug conditions.
- Because extra space is needed to store the debug condition values, additional processing overhead is added to the CPU and memory usage is increased. Thus, enabling crypto conditional debugging on a router with heavy traffic should be used with caution.
- Your router will perform conditional debugging only after at least one of the global crypto debug commands—**debug crypto isakmp**, **debug crypto ipsec**, or **debug crypto engine**—has been enabled. This requirement helps to ensure that the performance of the router will not be impacted when conditional debugging is not being used.

Enabling Crypto Conditional Debug Filtering

To enable crypto conditional debug filtering, perform the following tasks:.

	Command	Purpose
Step 1	Router# **enable**	Enables privileged EXEC mode.
Step 2	Router# **debug crypto condition** [**connid** *integer* **engine-id** *integer*] [**flowid** *integer* **engine-id** *integer*] [**fvrf** *string*] [**ivrf** *string*] [**peer** [**group** *string*] [**hostname** *string*] [**ipv4** ipaddress] [**subnet** *subnet mask*] [**username** *string*]] [**spi** *integer*] [**reset**]	Defines conditional debug filters. See Table 30-2 for descriptions of values.
Step 3	Router# **show crypto debug-condition** {[**peer**] [**connid**] [**spi**] [**fvrf**] [**ivrf**] [**unmatched**]}	Displays crypto debug conditions that have already been enabled in the router.
Step 4	Router# **debug crypto isakmp**	Enables global IKE debugging.
Step 5	Router# **debug crypto ipsec**	Enables global IPSec debugging.
Step 6	Router# **debug crypto engine**	Enables global crypto engine debugging.
Step 7	Router# **debug crypto condition unmatched** [**isakmp** | **ipsec** | **engine**]	(Optional) Displays debug conditional crypto messages when no context information is available to check against debug conditions. If none of the optional keywords are specified, all crypto-related information will be shown.

Disabling Crypto Conditional Debugging

Before you disable crypto conditional debugging, you must first disable any crypto global debug CLIs you have issued; thereafter, you can disable crypto conditional debugging. To disable crypto conditional debugging, enter the following command:

```
Router# debug crypto condition reset
```

Enabling Crypto Error Debug Messages

Enabling the **debug crypto error** command displays only error-related debug messages, thereby allowing you to easily determine why a crypto operation, such as an IKE negotiation, has failed within your system. To enable crypto error debug messages, enter the following command from privileged EXEC mode:

```
Router# debug crypto {isakmp | ipsec | engine} error
```

Note When enabling this command, ensure that global crypto debug commands are not enabled; otherwise, the global commands will override any possible error-related debug messages.

For complete configuration information for Crypto Conditional Debug Support, refer to this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_2/gt_dbcry.htm

Preparing for Online Insertion and Removal of a SPA

The Cisco 7600 series router supports online insertion and removal (OIR) of the SIP, in addition to each of the SPAs. Therefore, you can remove a SIP with its SPAs still intact, or you can remove a SPA independently from the SIP, leaving the SIP installed in the router.

This means that a SIP can remain installed in the router with one SPA remaining active, while you remove another SPA from one of the SIP subslots. If you are not planning to immediately replace a SPA into the SIP, then be sure to install a blank filler plate in the subslot. The SIP should always be fully installed with either functional SPAs or blank filler plates.

For more information about activating and deactivating SPAs in preparation for OIR, see the "Preparing for Online Insertion and Removal of SIPs and SPAs" topic in the "Troubleshooting a SIP" chapter in this guide.

PART 9

Field-Programmable Devices

CHAPTER 31

Upgrading Field-Programmable Devices

In general terms, field-programmable devices (FPDs) are hardware devices implemented on router cards that support separate upgrades. The term "FPD" has been introduced to collectively and generically describe any type of programmable hardware device on SIPs and SPAs. FPDs were introduced on the Cisco 7600 series router to support SPAs and SIPs.

This chapter describes the information that you need to know to verify image versions and to perform SIP and SPA FPD upgrades.

This chapter includes the following sections:

Release History

Table 31-1 provides the release and modification history for all FPD-related features on the Cisco 7600 series router.

Table 31-1 FPD Release History

Release	Modification
Cisco IOS Release 12.2(33)SRB	The **upgrade hw-module slot fpd file** command was introduced. This command replaces the **upgrade hw-module slot** command. The **upgrade hw-module subslot fpd file** command was introduced. This command replaces the **upgrade hw-module subslot** command
Cisco IOS Release 12.2(18)SXE	SIPs and SPAs were released on the Cisco 7600 series router and Catalyst 6500 series switch for the first time. FPD images were introduced to support these SPAs. The Fast Software Upgrade (FSU) procedure supported by Route Processor Redundancy (RPR) for supervisor engines was added to the documentation.

FPD Quick Upgrade

This section provides information if you simply want to upgrade FPDs for SIPs and SPAs as quickly as possible. These instructions are not always feasible for operating network environments and are not the only methods available for upgrading FPDs. If these methods of upgrade are not suitable for your situation, see the various other sections of this document for other methods of upgrading FPDs.

This section addresses the following topics:

- FPD Quick Upgrade Before Upgrading your Cisco IOS Release (Recommended), page 31-2
- FPD Quick Upgrade After Upgrading your Cisco IOS Release, page 31-2

FPD Quick Upgrade Before Upgrading your Cisco IOS Release (Recommended)

Step 1 When getting your Cisco IOS image, download the FPD image package for the Cisco IOS release that you are upgrading to any Flash disk on your router before booting the new version of Cisco IOS. The FPD image package can be retrieved from the same site where you went to get your Cisco IOS image. Do not change the name of the FPD image package.

Step 2 Boot using the new version of Cisco IOS. When the new Cisco IOS boots, it by default searches for the FPD image package in the router flash file systems and the FPD images will be updated automatically as part of the IOS boot process.

FPD Quick Upgrade After Upgrading your Cisco IOS Release

Step 1 An FPD upgrade is not always necessary after Cisco IOS is reloaded. If you have already reloaded your Cisco IOS, enter the **show hw-module all fpd** command to see if all system FPDs are compatible. If the FPDs are compatible, no further action is necessary. If at least one FPD needs an upgrade, proceed to Step 2.

Step 2 Go to the cisco.com site where you downloaded your specific Cisco IOS software and locate the FPD image package, if you haven't already.

Step 3 Download this FPD image package to a Flash disk on your router. Do not change the name of the FPD image package.

Do not change any FPD-related settings on your system (if **upgrade fpd auto** or **upgrade fpd path** has been changed, change the settings back to the default settings using the **no** form of the command). Reboot your Cisco IOS release software. When the new Cisco IOS boots, it by default searches for the FPD image package in the Flash file systems and the FPD images will be updated automatically as part of the IOS boot process.

Overview of FPD Images and Packages

An FPD image package is used to upgrade FPD images. Whenever a Cisco IOS image is released that supports carrier cards and SPAs, a companion FPD image package is also released for that Cisco IOS software release. The FPD image package is available from Cisco.com and is accessible from the Cisco Software Center page where you also go to download your Cisco IOS software image.

If you are running SIPs and SPAs on your router and are upgrading your Cisco IOS image, you should download the FPD image package file before booting the router using the new Cisco IOS release. If the SIP or SPA requires an FPD upgrade and the Cisco IOS image is unable to locate an FPD image package, the system messages will indicate that the FPD image is incompatible and you will need to go to the Cisco Software Center on Cisco.com to download the FPD image package for your Cisco IOS software release. An FPD incompatibility on a SPA disables all interfaces on that SPA until the incompatibility is addressed; an FPD incompatibility on a SIP disables all interfaces for all SPAs in the SIP until the incompatibility is addressed.

Note The FPD automatic upgrade feature only searches for the FPD image package file that is the same version number as the Cisco IOS release being used by the system. For example, if the Cisco IOS release being used is Cisco IOS Release 12.2(18)SXE, then the system will search for the FPD image package file that supports the specific Cisco IOS release (c7600-fpd-pkg.122-18.SXE.pkg). Therefore, ensure the FPD image package file on your system is compatible with your Cisco IOS release and do not change the name of the FPD image package file.

Upgrading FPD Images

This section documents some of the common scenarios where FPD image updates are necessary. It discusses the following scenarios:

- Migrating to a Newer Cisco IOS Release, page 31-3
- Upgrading FPD Images in a Production System, page 31-5

Migrating to a Newer Cisco IOS Release

This section discusses the following topics:

- Upgrading FPD Images Before Upgrading Cisco IOS Release (Recommended), page 31-3
- Upgrading FPD Images in a Production System, page 31-5
- Upgrading FPD Images Using Fast Software Upgrade, page 31-6

Upgrading FPD Images Before Upgrading Cisco IOS Release (Recommended)

If you are still running your old Cisco IOS Release but are preparing to load a newer version of Cisco IOS, you can upgrade FPD for the new Cisco IOS Release using the following method:

- Placing FPD Image Package on Flash Disk Before Upgrading IOS (Recommended), page 31-4

Placing FPD Image Package on Flash Disk Before Upgrading IOS (Recommended)

Placing the FPD image package for the IOS release that you are upgrading to before upgrading IOS is the recommended method for upgrading FPD because it is simple in addition to being fast. To perform this type of FPD upgrade, follow these steps:

Step 1 While still running the Cisco IOS release that will be upgraded, place the FPD image package for the new version of Cisco IOS onto one of your router's Flash file systems. For instance, if you are running Cisco IOS Release 12.2(18)SXE and are upgrading to Cisco IOS Release 12.2(19)SXE, place the FPD image package for Cisco IOS Release 12.2(19)SXE onto a Flash file system while still running Cisco IOS Release 12.2(18)SXE. You can locate the FPD image package for a specific IOS release on cisco.com from the same area where you download that Cisco IOS software image. Your router and SPAs should continue to operate normally since this action will have no impact on the current FPDs.

Caution Do not change the filename of the FPD image package file. The Cisco IOS searches for the FPD image package file by filename, so the FPD image package file cannot be found if it has been renamed.

Step 2 Reboot your router using the new upgraded Cisco IOS image. As part of the bootup process, the router will search for the FPD image package. Since the default settings for the FPD image package search are to check for the FPD image package for the specific Cisco IOS Release in a Flash file system, the FPD image package will be located during the bootup procedure and all FPDs that required upgrades will be upgraded.

Step 3 When the router has booted, verify the upgrade was successful by entering the **show hw-module all fpd** command.

Upgrade FPD Images after Upgrading the New Cisco IOS Release

The following steps explain how to upgrade FPD images if you have already upgraded your Cisco IOS release but still need to upgrade your FPD images.

To perform an FPD upgrade after the new Cisco release has been booted, follow these steps:

Step 1 If you are unsure if your FPD images for your SIPs and SPAs are compatible, enter the **show hw-module all fpd** command to verify compatibility of all SIPs and SPAs. If all of your SIPs and SPAs are compatible, there is no reason to perform this upgrade.

Step 2 If an FPD upgrade is necessary, place the FPD image package for the new version of Cisco IOS onto the router's Flash Disk or on an accessible FTP or TFTP server. You can locate the FPD image package on cisco.com from the same area where you downloaded your Cisco IOS software image.

Step 3 Enter the **upgrade hw-module [slot** *slot-number* **|** **subslot** *slot-number***/***subslot-number***]** *file-url* **[force]** command. The *file-url* command should direct users to the location of the FPD image package. For instance, if you had placed the FPD image package for Release 12.2(18)SXE on the TFTP server abrick/muck/myfolder, you would enter **upgrade hw-module [slot** *slot-number* **|** **subslot** *slot-number***/***subslot-number***]** **tftp://abrick/muck/myfolder/c7600-fpd-pkg.122-18.SXE.pkg** to complete this step.

If multiple SIPs or SPAs require upgrades, the different pieces of hardware will have to be updated individually.

Note the **force** option is used in this command. This option will force an FPD upgrade even if no FPD mismatch is detected. In instances where the **upgrade hw-module** command is entered, this option is almost never necessary and should only be entered if requested by a technical support representative.

Step 4 Verify the upgrade was successful by entering the **show hw-module all fpd** command.

Upgrading FPD Images in a Production System

Adding a SIP or SPA to a production system presents the possibility that the SIP or SPA may contain versions of FPD images that are incompatible with the Cisco IOS release currently running the router. In addition, the FPD upgrade operation can be a very CPU-intensive operation and therefore the upgrade operation may take more time when it is performed on a production system. The performance impact will vary depending on various factors, including network traffic load, the type of processing engine used, type of SPA, and the type of service configured.

For these reasons, we recommend that one of the following alternatives be used to perform the FPD upgrade on a production system if possible:

- Using a Non-Production System to Upgrade the SIP or SPA FPD Image, page 31-5
- Verifying System Compatibility First, page 31-6

Using a Non-Production System to Upgrade the SIP or SPA FPD Image

Before beginning the upgrade, ensure:

- The spare system is running the same version of the Cisco IOS software release that the target production system is running.
- The automatic upgrade feature is enabled on the spare system (the automatic upgrade feature is enabled by default. It can also be enabled using the **upgrade fpd auto** command).

Use the following procedure to perform an upgrade on a spare system:

Step 1 Download the FPD image package file to the router's flash file system or TFTP or FTP server accessible by the spare system. In most cases, it is preferable to place the file in a Flash file system since the router, by default, searches for the FPD image package in the Flash file systems. If the Flash file systems are full, use the **upgrade fpd path** command to direct the router to search for the FPD image package in the proper location.

Step 2 Insert the SIP or SPA into the spare system.

If an upgrade is required, the system will perform the necessary FPD image updates so that when this SIP or SPA is inserted to the target production system it will not trigger an FPD upgrade operation there.

Step 3 Verify the upgrade was successful by entering the **show hw-module all fpd** command.

Step 4 Remove the SIP or SPA from the spare system after the upgrade.

Step 5 Insert the SIP or SPA into the target production system.

Verifying System Compatibility First

If a spare system is not available to perform an upgrade, you can check for system compatibility by disabling the automatic upgrade feature before inserting the SIP or SPA (the automatic upgrade feature is enabled by default. It can be disabled using the **no upgrade fpd auto** command).

- If the FPD images on the SIP or SPA are compatible with the system, you will only need to re-enable the automatic upgrade feature (the automatic upgrade feature can be re-enabled using the **upgrade fpd auto** command).
- If the FPD images on the SIP or SPA are not compatible with the system, the SIP or SPA is disabled but will not impact system performance by attempting to perform an automatic upgrade.

Use the following procedure to check the FPD images on the SIP or SPA for system compatibility:

Step 1 Disable the automatic upgrade feature using the **no upgrade fpd auto** global configuration command.

Step 2 Insert the SIP or SPA into the system.

If the FPD images are compatible, the SIP or SPA will operate successfully after bootup.

If the FPD images are not compatible, the SIP or SPA is disabled. At this point we recommend that you wait for a scheduled maintenance when the system is offline to manually perform the FPD upgrade using one of the procedures outlined in the "Upgrading FPD Images" section on page 31-3.

Step 3 Re-enable the automatic upgrade feature using the **upgrade fpd auto** global configuration command.

Upgrading FPD Images Using Fast Software Upgrade

The fast software upgrade (FSU) procedure supported by Route Processor Redundancy (RPR) allows you to upgrade the Cisco IOS image on supervisor engines without reloading the system.

When using FSU to upgrade the Cisco IOS image, remember that Cisco IOS software is configured, by default, to automatically load the new FPD images from a flash file system on the router. Therefore, if the FPD image package for the new Cisco IOS has not been downloaded to the router flash file system, the FPD image that needs to be upgraded will not get upgraded if the new supervisor engine with the upgraded Cisco IOS becomes the primary supervisor engine. To ensure FPD is upgraded at the time of the FSU, place the FPD image package for the new version of Cisco IOS onto the flash file system before upgrading the Cisco IOS and follow the instructions in the "Upgrading FPD Images Before Upgrading Cisco IOS Release (Recommended)" section on page 31-3.

If a SIP or SPA is disabled after FSU is used to upgrade Cisco IOS and the supervisor engine with the upgraded Cisco IOS has become the primary supervisor engine, follow the instructions in the "Upgrade FPD Images after Upgrading the New Cisco IOS Release" section on page 31-4 to verify and, if necessary, upgrade FPD.

Optional FPD Procedures

This section provides information for optional FPD-related functions. None of the topics discussed in this section are necessary for completing FPD upgrades, but may be useful in some FPD-related scenarios. It covers the following topics:

Manually Upgrading SIP and SPA FPD Images

To manually upgrade the current FPD version on a SIP or SPA, use the following command:

```
Router# upgrade hw-module [slot slot-number | subslot slot-number/subslot-number] file
file-url [force]
```

In this example, *slot-number* is the slot where the SIP is installed, *subslot-number* is the subslot number where the SPA is located, *file-url* is the location and name of the FPD image package file, and **force** is an option that forces the SPA to perform an FPD upgrade even if FPD is compatible (the **force** option is almost never necessary and should only be entered if requested by a technical support representative). Note that **slot** *slot-number* is entered to specify a SIP FPD upgrade, while **subslot** *slot-number/subslot-number* is used to specify a SPA FPD upgrade. The SIP or SPA will automatically be reloaded to complete the FPD upgrade.

Caution An image upgrade can require a long period of time to complete depending on the SIP or SPA.

Upgrading FPD from an FTP or TFTP Server

The generally recommended method to perform an FPD image upgrade is to download the FPD image package to a Flash file system and use the FPD automatic upgrade. By default, the system searches the Flash file system for the FPD image package file when an FPD incompatibility is detected.

This default behavior of loading an FPD image from Flash can be changed using the **upgrade fpd path** global configuration command, which sets the path to search for the FPD image package file to a location other than the router's Flash file systems.

For large deployments where all the systems are being upgraded to a specific Cisco IOS software release, we recommend that the FPD image package file be placed on an FTP or TFTP server that is accessible to all the affected systems, and then use the **upgrade fpd path** global configuration command to configure the routers to look for the FPD image package file from the FTP or TFTP server prior to the reloading of the system with the new Cisco IOS release.

Note This approach can also be used if there is not enough disk space on the system Flash card to hold the FPD image package file.

To download an FPD image package file to an FTP or TFTP server, use the following procedure:

Step 1 Copy the FPD image package file to the FTP or TFTP server.

Step 2 Access the router from a connection that does not use the SPA interface for access, if possible. We recommend not using the SPA interface as your connection to the router because an FPD incompatibility disables all interfaces on the SPA, making a manual FPD upgrade impossible through a SPA interface.

If access through one of the SPA ports is the only access to the router you have, do not use the TFTP or FTP upgrade method. Instead, copy the FPD image package to your router's default Flash card before upgrading your Cisco IOS Release. This will allow the router to find the FPD image package during the first IOS bootup and upgrade FPD automatically.

Step 3 From global configuration mode, use the **upgrade fpd path** command to instruct the router to locate the FPD image package file from the FTP or TFTP server location.

For example, enter one of the following global configuration commands from the target system's console:

```
Router(config)# upgrade fpd path tftp://my_tftpserver/fpd_pkg_dir/
```

or

```
Router(config)# upgrade fpd path ftp://login:password@my_ftpserver/fpd_pkg_dir/
```

Note The final "/" at the end of each of the above examples is required. If the path is specified without the trailing "/" character, the command will not work properly.

In these examples, *my_tftpserver* or *my_ftpserver* is the path to server name, *fpd_pkg_dir* is the directory on the TFTP server where the FPD image package is located, and *login:password* is your FTP login name and password.

Step 4 Make sure that the FPD automatic upgrade feature is enabled by examining the output of the **show running-config** command. (Look for the *upgrade fpd auto* configuration line in the output. If there are no upgrade commands in the output, then **upgrade fpd auto** is enabled because it is the default setting.) If automatic upgrades are disabled, use the **upgrade fpd auto** global configuration command to enable automatic FPD upgrades.

Step 5 Enter the **show upgrade fpd file** command to ensure your router is connecting properly to the default FPD image package. If you are able to generate output related to the FPD image package using this command, the upgrade should work properly.

In the following example, the router is able to generate FPD image package information for the FPD image package on the TFTP server.

```
Router# show upgrade fpd file
tftp://mytftpserver/myname/myfpdpkg/c7600-fpd-pkg.122-18.SXE.pkg
Loading myname/myfpdpkg/c7600-fpd-pkg.122-18.SXE.pkg from 124.0.0.0 (via FastEthernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK]
Cisco Field Programmable Device Image Package for IOS
C7600 Family FPD Image Package (c7600-fpd-pkg.122-18.SXE.pkg), Version 12.2(SXE)
Copyright (c) 2004-2005 by cisco Systems, Inc.
Built Fri 25-Mar-2005 09:12 by integ


=============================== ================================================
                                        Bundled FPD Image Version Matrix
                                ================================================
                                                                   Min. Req.
Supported Card Types            ID  Image Name               Version  H/W Ver.
=============================== == ======================== ========= =========
2-port T3/E3 Serial SPA          1 T3E3 SPA ROMMON             2.12      0.0
                                 2 T3E3 SPA I/O FPGA           0.24      0.0
                                 3 T3E3 SPA E3 FPGA            0.6       0.0
                                 4 T3E3 SPA T3 FPGA            0.14      0.0
------------------------------- -- ------------------------ --------- ---------
```

```
4-port T3/E3 Serial SPA          1 T3E3 SPA ROMMON              2.12      0.0
                                 2 T3E3 SPA I/O FPGA            0.24      0.0
                                 3 T3E3 SPA E3 FPGA             0.6       0.0
                                 4 T3E3 SPA T3 FPGA             0.14      0.0
------------------------------- -- ------------------------- --------- ---------
...<additional output removed for readability>
```

Step 6 Save the configuration and reload the system with the new Cisco IOS release.

During the system startup after the reload, the necessary FPD image version check for all the SIPs and SPAs will be performed and any upgrade operation will occur automatically if an upgrade is required. In each upgrade operation, the system extracts the necessary FPD images to the SIP or SPA from the FPD image package file located on the FTP or TFTP server.

Modifying the Default Path for the FPD Image Package File Location

By default, the Cisco IOS software looks for the FPD image package file on a Flash file system when performing an automatic FPD image upgrade.

Note Be sure there is enough space on one of your Flash file systems to accommodate the FPD image package file.

Alternatively, you can store an FPD image package file elsewhere. However, because the system looks on the Flash file systems by default, you need to change the FPD image package file location so that the system is directed to search an alternate location (such an FTP or TFTP server) that is accessible by the Cisco IOS software. Enter the **upgrade fpd path** *fpd-pkg-dir-url* global configuration command, where *fpd-pkg-dir-url* is the alternate location, to instruct the router to search for the FPD image package elsewhere.

When specifying the *fpd-pkg-dir-url*, be aware of the following:

- The *fpd-pkg-dir-url* is the path to the FPD image package, but the FPD image package should not be specified as part of the *fpd-pkg-dir-url*. For instance, if the c7600-fpd-pkg.122-18.SXE.pkg file can be found on the TFTP server using the path mytftpserver/myname/myfpdpkg/c7600-fpd-pkg.122-18.SXE.pkg and you wanted the router to utilize this FPD image package for FPD upgrades, the **upgrade fpd path tftp://mytftpserver/myname/myfpdpkg/** command should be entered so the router knows where to find the file. The actual filename should not be specified.
- The final "/" character in the *fpd-pkg-dir-url* is required. In the preceding example, note that the *fpd-pkg-dir-url* is **tftp://mytftpserver/myname/myfpdpkg/.** Entering **tftp://mytftpserver/myname/myfpdpkg** (note: the final "/" character is missing) as the *fpd-pkg-dir-url* in that scenario would not work.

If the **upgrade fpd path** global configuration command has not been entered to direct the router to locate an FPD image package file in an alternate location, the system searches the Flash file systems on the Cisco 7600 series router for the FPD image package file.

Failure to locate an FPD image package file when an upgrade is required will disable the SIP or SPA. Because SIPs and SPAs will not come online until FPD is compatible, the SIP or SPA will also be disabled if it requires an FPD upgrade and the automatic upgrade feature is disabled.

Upgrading Multiple FPD Images

A single piece of hardware can contain multiple FPD images. The Cisco 7600 series router can upgrade up to 4 FPD images simultaneously. However, only one FPD upgrade per router slot can occur at a time, so all FPD images on all SIPs and SPAs in a single slot will have to wait for another FPD upgrade to finish.

Users should note that some FPD images require the SIP or SPA to reload to complete. The FPD upgrade process will perform this step automatically, so users do not have to intervene. However, the other FPDs in the hardware of the specified slot will have to wait for this reload to complete before their upgrade process begins.

During an automatic upgrade, the Cisco 7600 series router will upgrade as many FPDs as possible at a time. No user intervention is possible or necessary. The upgrade process will not stop until all FPD images have been updated.

During manual upgrades, it is important to note that users can only specify upgrades for a single piece of hardware each time the **upgrade hw-module** [**slot** *slot-number* | **subslot** *slot-number*/*subslot-number*] is entered. The up to 4 simultaneous upgrades applies to the manual upgrades as well; if you individually specify multiple manual FPD upgrades, only 4 FPDs can be upgraded simultaneously and that can only occur when the hardware is in different router slots. The FPD upgrade process will stop when all FPDs for the specified hardware have been upgraded.

Displaying Current and Minimum Required FPD Image Versions

To display the current version of FPD images on the SIPs and SPAs installed on your router, use the **show hw-module** [*slot-number*/*subslot-number* | **all**] **fpd** command, where *slot-number* is the slot number where the SIP is installed, and *subslot-number* is the number of the SIP subslot where a target SPA is located. Entering the **all** keyword shows information for hardware in all router slots.

The following examples show the output when using this **show** command.

The output display in this example shows that FPD versions on the SIPs and SPAs in the system meet the minimum requirements:

```
Router# show hw-module all fpd

==== ====================== ====== =============================================
                            H/W   Field Programmable   Current   Min. Required
Slot Card Type              Ver.  Device:"ID-Name"    Version     Version
==== ====================== ====== ================== =========== ==============
   1 7600-SIP-200           0.550 1-I/O FPGA            1.1         1.1
                                  2-EOS FPGA            1.211       1.211
                                  3-PEGASUS TX FPGA     1.129       1.129
                                  4-PEGASUS RX FPGA     1.3         1.3
                                  5-ROMMON              1.2         1.2
---- ---------------------- ------ ------------------ ----------- --------------
 1/1 SPA-2XOC3-ATM          0.225 1-I/O FPGA            1.24        1.24
---- ---------------------- ------ ------------------ ----------- --------------
   4 7600-SIP-200           0.550 1-I/O FPGA            1.1         1.1
                                  2-EOS FPGA            1.211       1.211
                                  3-PEGASUS TX FPGA     1.129       1.129
                                  4-PEGASUS RX FPGA     1.3         1.3
                                  5-ROMMON              1.2         1.2
---- ---------------------- ------ ------------------ ----------- --------------
 4/0 SPA-2XT3/E3            1.0   1-ROMMON              2.12        2.12
                                  2-I/O FPGA            0.24        0.24
                                  3-E3 FPGA             0.6         0.6
                                  4-T3 FPGA             0.14        0.14
---- ---------------------- ------ ------------------ ----------- --------------
```

```
 4/1 SPA-4XOC3-POS           0.209 1-I/O FPGA            3.4         3.4
---- ---------------------- ------ ------------------ ----------- --------------
 4/2 SPA-8XCHT1/E1           0.117 1-ROMMON              2.12        2.12
                                   2-I/O FPGA            1.2         1.2
==== ====================== ====== =============================================
```

This example shows the output when verifying all the FPDs for the carrier card and all the SPAs in a specific slot:

```
Router# show hw-module slot 4 fpd

==== ====================== ====== =============================================
                             H/W   Field Programmable   Current   Min. Required
Slot Card Type               Ver.  Device:"ID-Name"    Version      Version
==== ====================== ====== ================== =========== ==============
   4 7600-SIP-200            0.550 1-I/O FPGA            1.1         1.1
                                   2-EOS FPGA            1.211       1.211
                                   3-PEGASUS TX FPGA     1.129       1.129
                                   4-PEGASUS RX FPGA     1.3         1.3
                                   5-ROMMON              1.2         1.2
---- ---------------------- ------ ------------------ ----------- --------------
 4/0 SPA-2XT3/E3             1.0   1-ROMMON              2.12        2.12
                                   2-I/O FPGA            0.24        0.24
                                   3-E3 FPGA             0.6         0.6
                                   4-T3 FPGA             0.14        0.14
---- ---------------------- ------ ------------------ ----------- --------------
 4/1 SPA-4XOC3-POS           0.209 1-I/O FPGA            3.4         3.4
---- ---------------------- ------ ------------------ ----------- --------------
 4/2 SPA-8XCHT1/E1           0.117 1-ROMMON              2.12        2.12
                                   2-I/O FPGA            1.2         1.2
==== ====================== ====== =============================================
```

This example shows the output when using the *slot-number/subslot-number* argument to identify a particular SPA:

```
Router# show hw-module subslot 4/2 fpd

==== ====================== ====== =============================================
                             H/W   Field Programmable   Current   Min. Required
Slot Card Type               Ver.  Device:"ID-Name"    Version      Version
==== ====================== ====== ================== =========== ==============
 4/2 SPA-8XCHT1/E1           0.117 1-ROMMON              2.12        2.12
                                   2-I/O FPGA            1.2         1.2
==== ====================== ====== =============================================
```

The output display in this example shows that the SIP in slot 4 is disabled because one of the programmable devices does not meet the minimum version requirements. The output also contains a "NOTES" section that provides the name of the FPD image package file needed to upgrade the disabled SIP's FPD image.

```
Router#show hw-module all fpd

==== ====================== ====== =============================================
                             H/W   Field Programmable   Current   Min. Required
Slot Card Type               Ver.  Device:"ID-Name"    Version      Version
==== ====================== ====== ================== =========== ==============
   1 7600-SIP-200            0.550 1-I/O FPGA            1.1         1.1
                                   2-EOS FPGA            1.211       1.211
                                   3-PEGASUS TX FPGA     1.129       1.129
                                   4-PEGASUS RX FPGA     1.3         1.3
                                   5-ROMMON              1.2         1.2
```

```
---- --------------------- ------ ------------------ ----------- --------------
 1/1 SPA-2XOC3-ATM          0.225 1-I/O FPGA            1.24        1.24
---- --------------------- ------ ------------------ ----------- --------------
   4 7600-SIP... <DISABLED>  0.550 1-I/O FPGA            1.1         1.1
                                  2-EOS FPGA            1.211       1.211
                                  3-PEGASUS TX FPGA     1.129       1.129
                                  4-PEGASUS RX FPGA     1.3         1.3
                                  5-ROMMON              1.1         1.2     *
==== ===================== ====== ===========================================
 NOTES:
       - FPD images that are required to be upgraded are indicated with a '*'
         character in the "Minimal Required Version" field.
       - The following FPD image package file is required for the upgrade:
         "c7600-fpd-pkg.122-18.SXE.pkg"
```

Displaying Information About the Default FPD Image Package

You can use the **show upgrade fpd package default** command to find out which SIPs and SPAs are supported with your current Cisco IOS release and which FPD image package you need for an upgrade.

```
Router# show upgrade fpd package default

*************************************************************************
This IOS release requires the following default FPD Image Package for
the automatic upgrade of FPD images:
*************************************************************************

Version:12.2(SXE)

Package Filename:c7600-fpd-pkg.122-18.SXE.pkg

   List of card type supported in this package:

                                Minimal
         No. Card Type          HW Ver.
        ---- ------------------ -------
          1) 2 port adapter Enh   1.0
          2) 2xCT3 SPA            0.100
          3) 2xCT3 SPA            0.200
          4) 4xCT3 SPA            0.100
          5) 4xCT3 SPA            0.200

<additional output removed for readability>
```

Verifying the FPD Image Upgrade Progress

You can use the **show upgrade fpd progress** command to view a "snapshot" of the upgrade progress while an FPD image upgrade is taking place. The following example shows the type of information this command displays:

```
Router#show upgrade fpd progress

FPD Image Upgrade Progress Table:

 ==== ===================
====================================================
                                               Approx.
```

```
                             Field Programmable    Time      Elapsed
Slot Card Type               Device : "ID-Name"   Needed       Time    State
==== =================== ================== ========== ====================
 1/1 SPA-2XOC3-ATM       1-I/O FPGA           00:06:30   00:01:25 Updating...
---- ------------------- ------------------ ----------- --------------------
 4/0 SPA-2XT3/E3         1-ROMMON             00:00:30   00:00:02 Completed
                         2-I/O FPGA           00:01:00   00:00:01 Updating...
                         3-E3 FPGA            00:00:30   --:--:-- Waiting...
                         4-T3 FPGA            00:00:30   --:--:-- Waiting...
---- ------------------- ------------------ ----------- --------------------
 4/2 SPA-8XCHT1/E1       1-ROMMON             --:--:--   --:--:-- Waiting...
                         2-I/O FPGA           --:--:--   --:--:-- Waiting...
==== ==========================================================================
```

FPD Image Upgrade Examples

This section provides examples of automatic and manual FPD image upgrades. It includes the following examples:

- System Cannot Locate FPD Image Package File for an Automatic FPD Image Upgrade Example, page 31-13
- Automatic FPD Image Upgrade Example, page 31-13
- Manual FPD Image Upgrade Example, page 31-14
- Pending FPD Upgrade Example, page 31-15

System Cannot Locate FPD Image Package File for an Automatic FPD Image Upgrade Example

The following example displays the output when a SIP-200 requires an FPD upgrade and the **upgrade fpd auto** command is *enabled*, but the system cannot find the FPD image package file.

```
Mar 25 16:14:13:%FPD_MGMT-3-INCOMP_IMG_VER:Incompatible ROMMON (FPD ID=5) image version
detected for 7600-SIP-200 card in slot 1. Detected version = 1.1, minimum required version
= 1.2. Current HW version = 0.550.
Mar 25 16:14:13:%FPD_MGMT-5-UPGRADE_ATTEMPT:Attempting to automatically upgrade the FPD
image(s) for 7600-SIP-200 card in slot 1. Use 'show upgrade fpd progress' command to view
the upgrade progress ...
Mar 25 16:14:14:%FPD_MGMT-3-PKG_FILE_SEARCH_FAILED:FPD image package
(c7600-fpd-pkg.122-18.SXE.pkg) cannot be found in system's flash card or disk to do FPD
upgrade.
Mar 25 16:14:14:%OIR-6-REMCARD:Card removed from slot 1, interfaces disabled
Mar 25 16:14:14:%FPD_MGMT-5-CARD_DISABLED:7600-SIP-200 card in slot 1 is being disabled
because of an incompatible FPD image version. Note that the c7600-fpd-pkg.122-18.SXE.pkg
package will be required if you want to perform the upgrade operation.
Mar 25 16:14:14:%C6KPWR-SP-4-DISABLED:power to module in slot 1 set off (FPD Upgrade
Failed)
```

Automatic FPD Image Upgrade Example

The following example shows the output displayed when a SIP-200 requires an FPD image upgrade and the **upgrade fpd auto** command is *enabled*. In this example, the router has been configured to locate the FPD image package from a TFTP server, but most of the output would be similar regardless of the location of the FPD image package. The required FPD image is automatically upgraded.

```
Mar 25 16:22:48:%FPD_MGMT-3-INCOMP_IMG_VER:Incompatible ROMMON (FPD ID=5) image version
detected for 7600-SIP-200 card in slot 1. Detected version = 1.1, minimum required version
= 1.2. Current HW version = 0.550.
```

```
Mar 25 16:22:48:%FPD_MGMT-5-UPGRADE_ATTEMPT:Attempting to automatically upgrade the FPD
image(s) for 7600-SIP-200 card in slot 1. Use 'show upgrade fpd progress' command to view
the upgrade progress ...
Mar 25 16:22:48:%FPD_MGMT-6-BUNDLE_DOWNLOAD:Downloading FPD image bundle for 7600-SIP-200
card in slot 1 ...
Loading muck/luislu/c7600-fpd-pkg.122-18.SXE.pkg from 223.255.254.254 (via
GigabitEthernet5/1):!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Mar 25 16:23:17:%FPD_MGMT-6-UPGRADE_TIME:Estimated total FPD image upgrade time for
7600-SIP-200 card in slot 1 = 00:02:00.
Mar 25 16:23:17:%FPD_MGMT-6-UPGRADE_START:ROMMON (FPD ID=5) image upgrade in progress for
7600-SIP-200 card in slot 1. Updating to version 1.2. PLEASE DO NOT INTERRUPT DURING THE
UPGRADE PROCESS (estimated upgrade completion time = 00:02:00) ...
Mar 25 16:23:25:%FPD_MGMT-6-UPGRADE_PASSED:ROMMON (FPD ID=5) image in the 7600-SIP-200
card in slot 1 has been successfully updated from version 1.1 to version 1.2. Upgrading
time = 00:00:08.452
Mar 25 16:23:25:%FPD_MGMT-6-OVERALL_UPGRADE:All the attempts to upgrade the required FPD
images have been completed for 7600-SIP-200 card in slot 1. Number of successful/failure
upgrade(s):1/0.
Mar 25 16:23:26:%FPD_MGMT-5-CARD_POWER_CYCLE:7600-SIP-200 card in slot 1 is being power
cycled for the FPD image upgrade to take effect.
Mar 25 16:23:26:%OIR-6-REMCARD:Card removed from slot 1, interfaces disabled
Mar 25 16:23:26:%C6KPWR-SP-4-DISABLED:power to module in slot 1 set off (Reset)
Mar 25 16:24:16:%CWAN_RP-6-CARDRELOAD:Module reloaded on slot 1/0
Mar 25 16:24:18:%DIAG-SP-6-RUN_COMPLETE:Module 1:Running Complete Diagnostics...
Mar 25 16:24:18:%DIAG-SP-6-DIAG_OK:Module 1:Passed Online Diagnostics
Mar 25 16:24:19:%OIR-SP-6-INSCARD:Card inserted in slot 1, interfaces are now online
```

Manual FPD Image Upgrade Example

In the following example, FPD for the T1/E1 SPA in subslot 4/2 is upgraded manually from the FPD image package file that was placed on disk0:

```
Router# upgrade hw-module subslot 4/2 file disk0:c7600-fpd-pkg.122-18.SXE.pkg

% The following FPD(s) will be upgraded for SPA-8XCHT1/E1 (H/W ver = 0.117) in subslot
4/2:

    ================== =========== =========== ============
    Field Programmable   Current     Upgrade   Estimated
    Device:"ID-Name"    Version     Version   Upgrade Time
    ================== =========== =========== ============
    1-ROMMON               2.11        2.12      00:00:20
    2-I/O FPGA             1.1         1.2       00:01:00
    ================== =========== =========== ============


% Are you sure that you want to perform this operation? [no]:y
% Restarting the target card in subslot 4/2 for FPD image upgrade. Please wait ...

Router#
Mar 25 17:01:01:%FPD_MGMT-6-UPGRADE_TIME:Estimated total FPD image upgrade time for
SPA-8XCHT1/E1 card in subslot 4/2 = 00:01:20.
```

```
Mar 25 17:01:01:%FPD_MGMT-6-UPGRADE_START:ROMMON (FPD ID=1) image upgrade in progress for
SPA-8XCHT1/E1 card in subslot 4/2. Updating to version 2.12. PLEASE DO NOT INTERRUPT
DURING THE UPGRADE PROCESS (estimated upgrade completion time = 00:00:20) ...
Router#
Mar 25 17:01:04:%FPD_MGMT-6-UPGRADE_PASSED:ROMMON (FPD ID=1) image in the SPA-8XCHT1/E1
card in subslot 4/2 has been successfully updated from version 2.11 to version 2.12.
Upgrading time = 00:00:03.092
Mar 25 17:01:04:%FPD_MGMT-6-UPGRADE_START:I/O FPGA (FPD ID=2) image upgrade in progress
for SPA-8XCHT1/E1 card in subslot 4/2. Updating to version 1.2. PLEASE DO NOT INTERRUPT
DURING THE UPGRADE PROCESS (estimated upgrade completion time = 00:01:00) ...
Router#
Mar 25 17:01:26:%FPD_MGMT-6-UPGRADE_PASSED:I/O FPGA (FPD ID=2) image in the SPA-8XCHT1/E1
card in subslot 4/2 has been successfully updated from version 1.1 to version 1.2.
Upgrading time = 00:00:22.580
Mar 25 17:01:26:%FPD_MGMT-6-OVERALL_UPGRADE:All the attempts to upgrade the required FPD
images have been completed for SPA-8XCHT1/E1 card in subslot 4/2. Number of
successful/failure upgrade(s):2/0.
Router#
Mar 25 17:01:26:%FPD_MGMT-5-CARD_POWER_CYCLE:SPA-8XCHT1/E1 card in subslot 4/2 is being
power cycled for the FPD image upgrade to take effect.
```

Pending FPD Upgrade Example

In the following example, some FPD images are waiting for upgrades because the FPD upgrade process is upgrading another FPD on the same card (up to four FPD upgrades can occur at once, but the upgrades have to occur on hardware in different line card slots). In this particular example, the FPD upgrade process is happening on a SIP-200.

```
Mar 25 17:04:59:%FPD_MGMT-6-UPGRADE_TIME:Estimated total FPD image
upgrade time for 7600-SIP-200 card in slot 1 = 00:10:00.
Mar 25 17:04:59:%FPD_MGMT-6-UPGRADE_START:ROMMON (FPD ID=5) image
upgrade in progress for 7600-SIP-200 card in slot 1. Updating to version
1.2. PLEASE DO NOT INTERRUPT DURING THE UPGRADE PROCESS (estimated
upgrade completion time = 00:02:00) ...
Mar 25 17:05:08:%FPD_MGMT-6-UPGRADE_PASSED:ROMMON (FPD ID=5) image in
the 7600-SIP-200 card in slot 1 has been successfully updated from
version 1.1 to version 1.2. Upgrading time = 00:00:08.884
Mar 25 17:05:08:%FPD_MGMT-6-PENDING_UPGRADE:4 more FPD image upgrade
operation will be required on 7600-SIP-200 in slot 1 after additional
power-cycle operation on the target card.
Mar 25 17:05:08:%FPD_MGMT-5-CARD_POWER_CYCLE:7600-SIP-200 card in slot
1 is being power cycled for the FPD image upgrade to take effect.
Mar 25 17:05:08:%OIR-6-REMCARD:Card removed from slot 1, interfaces
disabled
Mar 25 17:05:08:%C6KPWR-SP-4-DISABLED:power to module in slot 1 set
off (Reset)
Mar 25 17:05:59:%CWAN_RP-6-CARDRELOAD:Module reloaded on slot 1/0
Mar 25 17:06:02:%FPD_MGMT-6-UPGRADE_TIME:Estimated total FPD image
upgrade time for 7600-SIP-200 card in slot 1 = 00:10:00.
Mar 25 17:06:02:%FPD_MGMT-6-UPGRADE_START:I/O FPGA (FPD ID=1) image
upgrade in progress for 7600-SIP-200 card in slot 1. Updating to version
1.1. PLEASE DO NOT INTERRUPT DURING THE UPGRADE PROCESS (estimated
upgrade completion time = 00:02:00) ...
Mar 25 17:06:21:%FPD_MGMT-6-UPGRADE_PASSED:I/O FPGA (FPD ID=1) image
in the 7600-SIP-200 card in slot 1 has been successfully updated from
version 1.0 to version 1.1. Upgrading time = 00:00:18.592
Mar 25 17:06:21:%FPD_MGMT-6-UPGRADE_START:EOS FPGA (FPD ID=2) image
upgrade in progress for 7600-SIP-200 card in slot 1. Updating to version
1.211. PLEASE DO NOT INTERRUPT DURING THE UPGRADE PROCESS (estimated
upgrade completion time = 00:02:00) ...
Mar 25 17:07:18:%FPD_MGMT-6-UPGRADE_PASSED:EOS FPGA (FPD ID=2) image
in the 7600-SIP-200 card in slot 1 has been successfully updated from
```

```
version 1.210 to version 1.211. Upgrading time = 00:00:56.812
Mar 25 17:07:18:%FPD_MGMT-6-UPGRADE_START:PEGASUS TX FPGA (FPD ID=3)
image upgrade in progress for 7600-SIP-200 card in slot 1. Updating to
version 1.129. PLEASE DO NOT INTERRUPT DURING THE UPGRADE PROCESS
(estimated upgrade completion time = 00:02:00) ...
Mar 25 17:08:17:%FPD_MGMT-6-UPGRADE_PASSED:PEGASUS TX FPGA (FPD ID=3)
image in the 7600-SIP-200 card in slot 1 has been successfully updated
from version 1.120 to version 1.129. Upgrading time = 00:00:59.188
Mar 25 17:08:17:%FPD_MGMT-6-UPGRADE_START:PEGASUS RX FPGA (FPD ID=4)
image upgrade in progress for 7600-SIP-200 card in slot 1. Updating to
version 1.3. PLEASE DO NOT INTERRUPT DURING THE UPGRADE PROCESS
(estimated upgrade completion time = 00:02:00) ...
Mar 25 17:09:03:%FPD_MGMT-6-UPGRADE_PASSED:PEGASUS RX FPGA (FPD ID=4)
image in the 7600-SIP-200 card in slot 1 has been successfully updated
from version 1.2 to version 1.3. Upgrading time = 00:00:45.396
Mar 25 17:09:03:%FPD_MGMT-6-OVERALL_UPGRADE:All the attempts to
upgrade the required FPD images have been completed for 7600-SIP-200
card in slot 1. Number of successful/failure upgrade(s):5/0.
Mar 25 17:09:03:%FPD_MGMT-5-CARD_POWER_CYCLE:7600-SIP-200 card in slot
1 is being power cycled for the FPD image upgrade to take effect.
```

Troubleshooting Problems with FPD Image Upgrades

This section contains information to help troubleshoot problems that can occur during the upgrade process.

Power Failure or Removal of a SIP or SPA During an FPD Image Upgrade

These instructions should only be used if a previous upgrade attempt has failed due to an external factor such as a power failure or a jacket card or SPA removal.

If the FPD upgrade operation is interrupted by a power failure or the removal of the SIP or SPA, it could corrupt the FPD image. This corruption of the FPD image file makes the SIP or SPA unusable by the router and the system will display the following messages when it tries to power up the SIP or SPA:

Note To find more information about FPD-related messages, check the system error messages guide for your Cisco IOS software release.

```
Mar 29 11:30:36:%SPA_OIR-3-RECOVERY_RELOAD:subslot 4/1:Attempting
recovery by reloading SPA
Mar 29 11:30:51:%SPA_OIR-3-HW_INIT_TIMEOUT:subslot 4/1
Mar 29 11:30:56:%SPA_OIR-3-RECOVERY_RELOAD:subslot 4/1:Attempting
recovery by reloading SPA
Mar 29 11:31:11:%SPA_OIR-3-HW_INIT_TIMEOUT:subslot 4/1
Mar 29 11:31:16:%SPA_OIR-3-RECOVERY_RELOAD:subslot 4/1:Attempting
recovery by reloading SPA
Mar 29 11:31:31:%SPA_OIR-3-HW_INIT_TIMEOUT:subslot 4/1
Mar 29 11:31:31:%SPA_OIR-3-SPA_POWERED_OFF:subslot 4/1:SPA 4xOC3 POS
SPA powered off after 5 failures within 600 seconds
```

The **show hw-module all fpd** command can be used to verify that the SIP or SPA is using a corrupted FPD image. In this example, the SPA in slot 4/1 is corrupted.

```
Router#show hw-module all fpd

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

```
                             H/W    Field Programmable   Current   Min. Required
Slot Card Type               Ver.   Device:"ID-Name"     Version      Version
==== ====================== ====== ================== =========== ==============
   4 7600-SIP-200            0.550 1-I/O FPGA             1.1         1.1
                                   2-EOS FPGA             1.211       1.211
                                   3-PEGASUS TX FPGA      1.129       1.129
                                   4-PEGASUS RX FPGA      1.3         1.3
                                   5-ROMMON               1.2         1.2
---- ---------------------- ------ ------------------ ----------- --------------
 4/1 SPA-4XOC3<DISABLED>  ?.?   ????????????        ?.?         ?.?
==== ====================== ====== ============================================
```

Performing a FPD Recovery Upgrade

The recovery upgrade procedure can only be performed on a SIP or SPA that has been powered off by the system after it has failed all of the retries attempted to initialize the SIP or SPA.

The following example displays the output of an attempt to perform a recovery upgrade before all the initialization retries have been attempted for the SPA in subslot 4/1.

Note Other factors can cause the system to ask "Do you want to perform the recovery upgrade operation?" Only answer **y** to this question if you have attempted an FPD upgrade that has failed due to a power failure or a SIP or SPA removal.
If you are prompted for this question without having previously had a failed upgrade attempt for one of the aforementioned reasons, contact Cisco Technical Support.

```
Mar 29 11:29:55:%SPA_OIR-3-RECOVERY_RELOAD:subslot 4/1:Attempting
recovery by reloading SPA
Mar 29 11:30:10:%SPA_OIR-3-HW_INIT_TIMEOUT:subslot 4/1
Mar 29 11:30:15:%SPA_OIR-3-RECOVERY_RELOAD:subslot 4/1:Attempting
recovery by reloading SPA
Mar 29 11:30:31:%SPA_OIR-3-HW_INIT_TIMEOUT:subslot 4/1
Router#upgrade hw-module subslot 4/1 file
disk0:c7600-fpd-pkg.122-18.SXE.pkg

% Cannot get FPD version information for version checking. If a previous
upgrade attempt has failed for the target card, then a recovery upgrade
would be required to fix the failure.

% The following FPD(s) will be upgraded for SPA-4XOC3-POS (H/W ver =
0.209) in subslot 4/1:

    ================== =========== =========== ============
    Field Programmable   Current     Upgrade   Estimated
    Device:"ID-Name"    Version     Version   Upgrade Time
    ================== =========== =========== ============
    1-I/O FPGA             ?.?         3.4       00:02:00
    ================== =========== =========== ============


% Do you want to perform the recovery upgrade operation? [no]:y
% Cannot perform recovery upgrade operation because the target card is
not in a failed state. Please try again later.
```

Once the following error message is displayed, you can perform the recovery upgrade:

Note You must wait to see this error message before you attempt the upgrade.

```
Mar 29 11:31:31:%SPA_OIR-3-SPA_POWERED_OFF:subslot 4/1:SPA 4xOC3 POS SPA powered off after
5 failures within 600 seconds
```

Perform the manual FPD image upgrade method using the **upgrade hw-module subslot** command to recover from a corrupted image after the SIP or SPA has been powered off by the system. In this command, *slot-number* is the slot where the SIP is installed, *subslot-number* is the subslot of the SIP where the SPA is located, and *file-url* is the location of the FPD image package file.

Note Before proceeding with this operation, make sure that the correct version of the FPD image package file has been obtained for the corresponding Cisco IOS release that the system is using.

The following example displays the console output of a recovery upgrade operation:

```
Router#upgrade hw-module subslot 4/1 file
disk0:c7600-fpd-pkg.122-18.SXE.pkg

% Cannot get FPD version information for version checking. If a previous
upgrade attempt has failed for the target card, then a recovery upgrade
would be required to fix the failure.


% The following FPD(s) will be upgraded for SPA-4XOC3-POS (H/W ver =
0.209) in subslot 4/1:

    ================== =========== =========== ============
    Field Programmable   Current     Upgrade   Estimated
    Device:"ID-Name"    Version     Version   Upgrade Time
    ================== =========== =========== ============
    1-I/O FPGA             ?.?         3.4       00:02:00
    ================== =========== =========== ============


% Do you want to perform the recovery upgrade operation? [no]:y
% Proceeding with recovery upgrade operation ...

Router#
Mar 29 11:37:51:%FPD_MGMT-6-UPGRADE_TIME:Estimated total FPD image
upgrade time for SPA-4XOC3-POS card in subslot 4/1 = 00:02:00.
Mar 29 11:37:51:%FPD_MGMT-6-UPGRADE_START:Unknown FPD (FPD ID=1) image
upgrade in progress for SPA-4XOC3-POS card in subslot 4/1. Updating to
version 3.4. PLEASE DO NOT INTERRUPT DURING THE UPGRADE PROCESS
(estimated upgrade completion time = 00:02:00) ...
Router#
Mar 29 11:39:11:%FPD_MGMT-6-UPGRADE_PASSED:Unknown FPD (FPD ID=1)
image in the SPA-4XOC3-POS card in subslot 4/1 has been successfully
updated from version ?.? to version 3.4. Upgrading time = 00:01:19.528
Mar 29 11:39:11:%FPD_MGMT-6-OVERALL_UPGRADE:All the attempts to
upgrade the required FPD images have been completed for SPA-4XOC3-POS
card in subslot 4/1. Number of successful/failure upgrade(s):1/0.
Mar 29 11:39:11:%FPD_MGMT-5-CARD_POWER_CYCLE:SPA-4XOC3-POS card in
subslot 4/1 is being power cycled for the FPD image upgrade to take
effect.
```

Verifying a Successful Upgrade

After the upgrade process is complete, you can use the **show hw-module all fpd** command to verify that the FPD image has been successfully upgraded:

```
Router#show hw-module all fpd

==== ====================== ====== =============================================
                             H/W   Field Programmable   Current   Min. Required
Slot Card Type               Ver.  Device:"ID-Name"    Version      Version
==== ====================== ====== ================== =========== ==============
   4 7600-SIP-200            0.550 1-I/O FPGA             1.1         1.1
                                   2-EOS FPGA             1.211       1.211
                                   3-PEGASUS TX FPGA      1.129       1.129
                                   4-PEGASUS RX FPGA      1.3         1.3
                                   5-ROMMON               1.2         1.2
---- ---------------------- ------ ------------------ ----------- --------------
4/1 SPA-4XOC3-POS           0.209  1-I/O FPGA             3.4         3.4
==== ====================== ====== =============================================
```

PART 10

Glossary

GLOSSARY

B

blank filler plate An empty panel used to fill vacant subslots on a SIP. For proper operation, a SIP should be fully installed with either functional SPAs or blank filler plates.

D

double height Describes the dimension of a SPA that occupies two, vertically-aligned SIP subslots.

F

FPD Field-programmable device. General term for any hardware component implemented on router cards that supports separate software upgrades. SIPs and SPAs must have the right FPD version to function properly; an FPD incompatibility will disable all interfaces on the SPA or all SPAs within the SIP.

FPD image package An FPD image package is used to upgrade FPD images. Whenever a Cisco IOS image is released that supports SPAs, a companion SPA FPD image package is also released for that Cisco IOS software release.

O

OIR Online insertion and removal. Feature supported by SIPs and SPAs allowing removal of the cards while the router and the cards are activated, without affecting the operation of other cards or the router. Although this removal can be done while the SIP or SPA is activated, it is generally recommended that you gracefully deactivate the hardware using the appropriate commands for your platform prior to removal of the hardware.

S

SFP Small form-factor pluggable optical transceiver. A type of fiber optic receptacle device that mounts flush with the front panel to provide network connectivity.

single height Describes the dimension of a SPA that occupies a single SIP subslot, or half of the SIP.

SIP SPA interface processor. A SIP is a platform-specific carrier card that inserts into a router slot like a line card. A SIP can hold one or more SPAs in its subslots, depending on the SIP type. The SPA provides the network interface. The SIP provides the connection between the route processor (RP) and the SPA.

SPA Shared port adapter. A SPA is a modular, platform-independent port adapter that inserts into a subslot of a compatible SIP carrier card to provide network connectivity and increased interface port density. The SPA provides the interface between the network and the SIP.

subslot Secondary slot on a SIP where a SPA is installed. The primary slot is the chassis slot on the router.

INDEX

Symbols

Numerics

A

B

C

D

E

F

G

H

K

L

M

N

O

P

Q

R

S

T

U

V

W

X