MPLS Basic MPLS Configuration Guide, Cisco IOS Release 12.4T

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

CONTENTS

Multiprotocol Label Switching Overview

This chapter describes the Multiprotocol Label Switching (MPLS) distribution protocol. MPLS is a high-performance packet forwarding technology that integrates the performance and traffic management capabilities of data link layer (Layer 2) switching with the scalability, flexibility, and performance of network-layer (Layer 3) routing. It enables service providers to meet challenges brought about by explosive growth and provides the opportunity for differentiated services without necessitating the sacrifice of existing infrastructure.

The MPLS architecture is remarkable for its flexibility:

- Data can be transferred over any combination of Layer 2 technologies
- Support is offered for all Layer 3 protocols
- Scaling is possible well beyond anything offered in today's networks.

Specifically, MPLS can efficiently enable the delivery of IP services over an ATM switched network. It supports the creation of different routes between a source and a destination on a purely router-based Internet backbone. Service providers who use MPLS can save money and increase revenue and productivity.

Note Label switching on a router requires that Cisco Express Forwarding be enabled on that router. Refer to the Cisco Express Forwarding feature documentation for configuration information.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information

about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

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

MPLS Tag Switching Terminology

Beginning with Cisco IOS Release 12.1, the Tag Switching distribution protocol has been replaced with the MPLS distribution protocol. The Tag Switching command-line interface (CLI) commands are supported but will be discontinued in a future release.

The table below lists tag switching terms (found in earlier releases of this document) and the equivalent MPLS terms used in this document.

Table 1 Equivalency Table for Tag Switching and MPLS Terms

Old Tag Switching Terminology	New MPLS Terminology
Tag Switching	Multiprotocol Label Switching (MPLS)
Tag (short for Tag Switching)	MPLS
Tag (item or packet)	Label
TDP (Tag Distribution Protocol)	LDP (Label Distribution Protocol) Cisco TDP and LDP (MPLS Label Distribution Protocol) are nearly identical in function, but use incompatible message formats and some different procedures. Cisco is changing from TDP to a fully compliant LDP.
Tag Switched	Label Switched
TFIB (Tag Forwarding Information Base)	LFIB (Label Forwarding Information Base)
TSR (Tag Switching Router)	LSR (Label Switching Router)
TSC (Tag Switch Controller)	LSC (Label Switch Controller)
ATM-TSR (ATM Tag Switch Router)	ATM-LSR (ATM Label Switch Router, such as the Cisco BPX 8650 switch)
TVC (Tag VC, Tag Virtual Circuit)	LVC (Label VC, Label Virtual Circuit)
TSP (Tag Switch Path)	LSP (Label Switch Path)
XTag ATM (extended Tag ATM port)	XmplsATM (extended MPLS ATM port)

MPLS Commands and Saved Configurations

During the transition period from tag switching to MPLS, if a configuration command has both MPLS and tag switching forms, the tag switching version is written to saved configurations. For example, you can configure MPLS hop-by-hop forwarding for a router POS interface by issuing the following commands:

```
Router# configure terminal
Router(config)# interface POS3/0
Router(config-if)# mpls ip
```

In this example, the **mplsip** command has a tag switching form (**tag-switchingip**). After you enter these commands and save this configuration or display the running configuration by means of the **showrunningconfiguration**command, the configuration commands appear as follows:

```
interface POS3/0
tag-switching ip
```

Saving the tag switching form of commands (that have both tag switching and MPLS forms) allows for backward compatibility. You can use a new router software image to modify and write configurations, and then later use configurations created by the new image with earlier software versions that do not support the MPLS forms of commands

Using the tag switching forms of the commands allows older software that supports tag switching commands, but not new MPLS commands, to successfully interpret interface configurations.

MPLS Tag Switching CLI Command Summary

The table below summarizes general-purpose MPLS commands. Except where otherwise noted, these MPLS commands have been derived from existing tag-switching commands to preserve the familiar syntax of existing commands that formed the basis for implementing new MPLS functionality. The tag-switching versions of the command will be discontinued in a future release.

Table 2 Summary of MPLS Commands Described in this Document

Command	Corresponding Tag Switching Command	Description
debug mpls adjacency	**debug tag-switching adjacency**	Displays changes to label switching entries in the adjacency database.
debug mpls events	**debug tag-switching events**	Displays information about significant MPLS events.
debug mpls lfib cef	**debug tag-switching tfib cef**	Prints detailed information about label rewrites being created, resolved, and deactivated as CEF routes are added, changed, or removed.
debug mpls lfib enc	**debug tag-switching tfib enc**	Prints detailed information about label encapsulations while label rewrites are created or updated and placed into the label forwarding information base (LFIB).
debug mpls lfib lsp	**debug tag-switching tfib tsp**	Prints detailed information about label rewrites being created and deleted as TSP tunnels are added or removed.

Command	Corresponding Tag Switching Command	Description
debug mpls lfib state	**debug tag-switching tfib state**	Traces what happens when label switching is enabled or disabled.
debug mpls lfib struct	**debug tag-switching tfib struct**	Traces the allocation and freeing of LFIB-related data structures, such as the LFIB itself, label-rewrites, and label-info data.
debug mpls packets	**debug tag-switching packets**	Displays labeled packets switched by the host router.
interface atm	**interface atm**	Enters interface configuration mode, specifies ATM as the interface type, and enables the creation of a subinterface on the ATM interface.
mpls atm control-vc	**tag-switching atm control-vc**	Configures the VPI and VCI to be used for the initial link to the label switching peer device.
mpls atm vpi	**tag-switching atm vpi**	Configures the range of values to be used in the VPI field for label VCs.
mpls ip (global configuration)	**tag-switching ip (global configuration)**	Enables MPLS forwarding of IPv4 packets along normally routed paths for the platform.
mpls ip (interface configuration)	**tag-switching ip (interface configuration)**	Enables MPLS forwarding of IPv4 packets along normally routed paths for a particular interface.
mpls ip default-route	**tag-switching ip default-route**	Enables the distribution of labels associated with the IP default route.
mpls ip propagate-ttl	**tag-switching ip propagate-ttl**	Sets the time-to-live (TTL) value when an IP packet is encapsulated in MPLS.
mpls ip ttl-expiration pop	N/A	Forwards packets using the global IP routing table or the original label stack, depending on the number of labels in the packet.
mpls label range	**tag-switching tag-range downstream**	Configures the range of local labels available for use on packet interfaces. **Note** The syntax of this command differs slightly from its tag-switching counterpart.
mpls mtu	**tag-switching mtu**	Sets the per-interface maximum transmission unit (MTU) for labeled packets.

Command	Corresponding Tag Switching Command	Description
show mpls forwarding-table	**show tag-switching forwarding-table**	Displays the contents of the label forwarding information base (LFIB).
show mpls interfaces	**show tag-switching interfaces**	Displays information about one or more interfaces that have been configured for label switching.
show mpls label range	N/A	Displays the range of local labels available for use on packet interfaces.

Benefits

MPLS provides the following major benefits to service provider networks:

- Scalable support for Virtual Private Networks (VPNs)--MPLS enables VPN services to be supported in service provider networks, thereby greatly accelerating Internet growth.

The use of MPLS for VPNs provides an attractive alternative to the building of VPNs by means of either ATM or Frame Relay permanent virtual circuits (PVCs) or various forms of tunneling to interconnect routers at customer sites.

Unlike the PVC VPN model, the MPLS VPN model is highly scalable and can accommodate increasing numbers of sites and customers. The MPLS VPN model also supports “any-to-any” communication among VPN sites without requiring a full mesh of PVCs or the backhauling (suboptimal routing) of traffic across the service provider network. For each MPLS VPN user, the network of the service provider appears to function as a private IP backbone over which the user can reach other sites within the VPN organization, but not the sites of any other VPN organization.

From a user perspective, the MPLS VPN model enables network routing to be dramatically simplified. For example, rather than needing to manage routing over a topologically complex virtual backbone composed of many PVCs, an MPLS VPN user can generally employ the backbone of the service provider as the default route in communicating with all of the other VPN sites.

- Explicit routing capabilities (also called constraint-based routing or traffic engineering)--Explicit routing employs “constraint-based routing,” in which the path for a traffic flow is the shortest path that meets the resource requirements (constraints) of the traffic flow.

In MPLS traffic engineering, factors such as bandwidth requirements, media requirements, and the priority of one traffic flow versus another can be taken into account. These traffic engineering capabilities enable the administrator of a service provider network to perform the following tasks:

- ◦ Control traffic flow in the network
  ◦ Reduce congestion in the network
  ◦ Make best use of network resources

Thus, the network administrator can specify the amount of traffic expected to flow between various points in the network (thereby establishing a traffic matrix), while relying on the routing system to perform the following tasks:

- ◦ Calculate the best paths for network traffic
  ◦ Set up the explicit paths to carry the traffic

Label Switching Functions

In conventional Layer 3 forwarding mechanisms, as a packet traverses the network, each router extracts all the information relevant to forwarding the packet from the Layer 3 header. This information is then used as an index for a routing table lookup to determine the next hop for the packet.

In the most common case, the only relevant field in the header is the destination address field, but in some cases, other header fields might also be relevant. As a result, the header analysis must be done independently at each router through which the packet passes. In addition, a complicated table lookup must also be done at each router.

In label switching, the analysis of the Layer 3 header is done only once. The Layer 3 header is then mapped into a fixed length, unstructured value called a *label* .

Many different headers can map to the same label, as long as those headers always result in the same choice of next hop. In effect, a label represents a *forwarding equivalence class* --that is, a set of packets which, however different they may be, are indistinguishable by the forwarding function.

The initial choice of a label need not be based exclusively on the contents of the Layer 3 packet header; for example, forwarding decisions at subsequent hops can also be based on routing policy.

Once a label is assigned, a short label header is added at the front of the Layer 3 packet. This header is carried across the network as part of the packet. At subsequent hops through each MPLS router in the network, labels are swapped and forwarding decisions are made by means of MPLS forwarding table lookup for the label carried in the packet header. Hence, the packet header does not need to be reevaluated during packet transit through the network. Because the label is of fixed length and unstructured, the MPLS forwarding table lookup process is both straightforward and fast.

Distribution of Label Bindings

Each> label switching router (LSR) in the network makes an independent, local decision as to which label value to use to represent a forwarding equivalence class. This association is known as a label binding. Each LSR informs its neighbors of the label bindings it has made. This awareness of label bindings by neighboring routers is facilitated by the following protocols:

- Tag Distribution Protocol (TDP)--Used to support MPLS forwarding along normally routed paths
- Resource Reservation Protocol (RSVP)--Used to support MPLS traffic engineering
- Border Gateway Protocol (BGP)--Used to support MPLS virtual private networks (VPNs)

When a labeled packet is being sent from LSR A to the neighboring LSR B, the label value carried by the IP packet is the label value that LSR B assigned to represent the forwarding equivalence class of the packet. Thus, the label value changes as the IP packet traverses the network.

MPLS and Routing

A label represents a forwarding equivalence class, but it does not represent a particular path through the network. In general, the path through the network continues to be chosen by the existing Layer 3 routing algorithms such as OSPF, Enhanced IGRP, and BGP. That is, at each hop when a label is looked up, the next hop chosen is determined by the dynamic routing algorithm.

MPLS Traffic Engineering

MPLS traffic engineering software enables an MPLS backbone to replicate and expand upon the traffic engineering capabilities of Layer 2 ATM and Frame Relay networks. MPLS is an integration of Layer 2 and Layer 3 technologies. By making traditional Layer 2 features available to Layer 3, MPLS enables traffic engineering. Thus, you can offer in a one-tier network what now can be achieved only by overlaying a Layer 3 network on a Layer 2 network.

Traffic engineering is essential for service provider and Internet service provider (ISP) backbones. Such backbones must support a high use of transmission capacity, and the networks must be very resilient so that they can withstand link or node failures.

MPLS traffic engineering provides an integrated approach to traffic engineering. With MPLS, traffic engineering capabilities are integrated into Layer 3, which optimizes the routing of IP traffic, given the constraints imposed by backbone capacity and topology.

- Why Use MPLS Traffic Engineering, page 7
- How MPLS Traffic Engineering Works, page 7

Why Use MPLS Traffic Engineering

WAN connections are an expensive item in an ISP budget. Traffic engineering enables ISPs to route network traffic to offer the best service to their users in terms of throughput and delay. By making the service provider more efficient, traffic engineering reduces the cost of the network.

Currently, some ISPs base their services on an overlay model. In the overlay model, transmission facilities are managed by Layer 2 switching. The routers see only a fully meshed virtual topology, making most destinations appear one hop away. If you use the explicit Layer 2 transit layer, you can precisely control how traffic uses available bandwidth. However, the overlay model has numerous disadvantages. MPLS traffic engineering achieves the traffic engineering benefits of the overlay model without running a separate network, and without needing a nonscalable, full mesh of router interconnects.

How MPLS Traffic Engineering Works

MPLS traffic engineering automatically establishes and maintains LSPs across the backbone by using RSVP. The path that an LSP uses is determined by the LSP resource requirements and network resources, such as bandwidth.

Available resources are flooded by means of extensions to a link-state-based Interior Gateway Protocol (IGP).

Traffic engineering tunnels are calculated at the LSP head based on a fit between required and available resources (constraint-based routing). The IGP automatically routes the traffic onto these LSPs. Typically, a packet crossing the MPLS traffic engineering backbone travels on a single LSP that connects the ingress point to the egress point.

MPLS traffic engineering is built on the following Cisco IOS mechanisms:

- IP tunnel interfaces--From a Layer 2 standpoint, an MPLS tunnel interface represents the head of an LSP. It is configured with a set of resource requirements, such as bandwidth and media requirements, and priority.

From a Layer 3 standpoint, an LSP tunnel interface is the head-end of a unidirectional virtual link to the tunnel destination.

- MPLS traffic engineering path calculation module--This calculation module operates at the LSP head. The module determines a path to use for an LSP. The path calculation uses a link-state database containing flooded topology and resource information.
- RSVP with traffic engineering extensions--RSVP operates at each LSP hop and is used to signal and maintain LSPs based on the calculated path.
- MPLS traffic engineering link management module--This module operates at each LSP hop, does link call admission on the RSVP signalling messages, and does bookkeeping of topology and resource information to be flooded.
- Link-state IGP (Intermediate System-to-Intermediate System (IS-IS) or OSPF--each with traffic engineering extensions)--These IGPs are used to globally flood topology and resource information from the link management module.
- Enhancements to the SPF calculation used by the link-state IGP (IS-IS or OSPF)--The IGP automatically routes traffic onto the appropriate LSP tunnel based on tunnel destination. Static routes can also be used to direct traffic onto LSP tunnels.
- Label switching forwarding--This forwarding mechanism provides routers with a Layer 2-like ability to direct traffic across multiple hops of the LSP established by RSVP signalling.

One approach to engineering a backbone is to define a mesh of tunnels from every ingress device to every egress device. The MPLS traffic engineering path calculation and signalling modules determine the path taken by the LSPs for these tunnels, subject to resource availability and the dynamic state of the network. The IGP, operating at an ingress device, determines which traffic should go to which egress device, and steers that traffic into the tunnel from ingress to egress.

A flow from an ingress device to an egress device might be so large that it cannot fit over a single link, so it cannot be carried by a single tunnel. In this case, multiple tunnels between a given ingress and egress can be configured, and the flow is load-shared among them.

MPLS Virtual Private Networks

Using MPLS VPNs in a Cisco IOS network provide the capability to deploy and administer scalable Layer 3 VPN backbone services including applications, data hosting network commerce, and telephony services to business customers. A VPN is a secure IP-based network that shares resources on one or more physical networks. A VPN contains geographically dispersed sites that can communicate securely over a shared backbone.

A one-to-one relationship does not necessarily exist between customer sites and VPNs; a given site can be a member of multiple VPNs. However, a site can associate with only one VPN routing and forwarding instance (VRF). Each VPN is associated with one or more VPN VRFs. A VRF includes routing and forwarding tables and rules that define the VPN membership of customer devices attached to CE routers. A VRF consists of the following:

- IP routing table
- CEF table
- Set of interfaces that use the CEF forwarding table
- Set of rules and routing protocol parameters to control the information in the routing tables

VPN routing information is stored in the IP routing table and the CEF table for each VRF. A separate set of routing and CEF tables is maintained for each VRF. These tables prevent information from being forwarded outside a VPN and also prevent packets that are outside a VPN from being forwarded to a router within the VPN.

MPLS Quality of Service

The quality of service (QoS) feature for MPLS enables network administrators to provide differentiated types of service across an MPLS network. Differentiated service satisfies a range of requirements by supplying for each packet transmitted the particular kind of service specified for that packet by its QoS. Service can be specified in different ways, for example, using the IP precedence bit settings in IP packets.

In supplying differentiated service, MPLS QoS offers packet classification, congestion avoidance, and congestion management. The table below lists these functions and their descriptions.

Table 3 QoS Services and Features

Service	QoS Function	Description
Packet classification	Committed access rate (CAR). Packets are classified at the edge of the network before labels are assigned.	Classifies packets according to input or output transmission rates. Allows you to set the MPLS experimental bits or the IP Precedence or DSCP bits (whichever is appropriate).
Congestion avoidance	Weighted Random Early Detection (WRED). Packet classes are differentiated based on drop probability.	Monitors network traffic to prevent congestion by dropping packets based on the IP Precedence or DSCP bits or the MPLS experimental field.
Congestion management	Class-based weighted fair queueing (CBWFQ). Packet classes are differentiated based on bandwidth and bounded delay.	An automated scheduling system that uses a queueing algorithm to ensure bandwidth allocation to different classes of network traffic.

Note MPLS QoS lets you duplicate Cisco IOS IP QoS (Layer 3) features as closely as possible in MPLS devices, including label edge routers (LERs), LSRs, and ATM-LSRs. MPLS QoS functions map nearly one-for-one to IP QoS functions on all interface types.

For more information on configuration of the QoS functions (CAR, WRED, and CBWFQ), refer to the *Cisco IOS Quality of Service Solutions Configuration Guide* .

For complete command syntax information for CAR, WRED, and WFQ, refer to the *Cisco IOS Quality of Service Solutions Command Reference.*

- Specifying the QoS in the IP Precedence Field, page 9

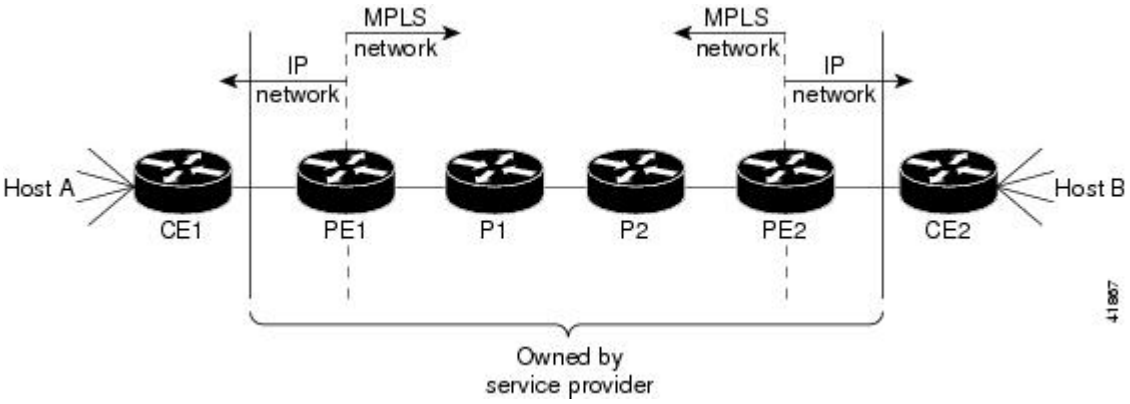
Specifying the QoS in the IP Precedence Field

When you send IP packets from one site to another, the IP Precedence field (the first three bits of the DSCP field in the header of an IP packet) specifies the QoS. Based on the IP precedence marking, the packet is given the desired treatment such as the latency or the percent of bandwidth allowed for that quality of service. If the service provider network is an MPLS network, then the IP precedence bits are copied into the

MPLS EXP field at the edge of the network. However, the service provider might want to set a QoS for a MPLS packet to a different value determined by the service offering.

This feature allows the service provider to set the MPLS experimental field instead of overwriting the value in the IP precedence field belonging to a customer. The IP header remains available for the customer's use; the QoS of an IP packet is not changed as the packet travels through the MPLS network.

The figure below shows an MPLS network that connects two sites of a IP network belonging to a customer.



Note The network is bidirectional, but for the purpose of this document the packets move left to right.

In the figure above, the symbols have the following meanings displayed in the table below:

Table 4 Device Symbols

Symbol	Meaning
CE1	Customer equipment 1
PE1	Service provider edge router (ingress LSR)
P1	Service provider router within the core of the network of the service provider
P2	Service provider router within the core of the network of the service provider
PE2	Service provider edge router (egress LSR)
CE2	Customer equipment 2

Note Notice that PE1 and PE2 are at the boundaries between the MPLS network and the IP network.

In the figure above, the following behavior occurs:

- Packets arrive as IP packets at PE1, the provider edge router (also known as the ingress label switching router).
- PE1 sends the packets as MPLS packets.

- Within the service provider network, there is *no IP Precedence field* for the queueing mechanism to look at because the packets are MPLS packets. The packets remain MPLS packets until they arrive at PE2, the provider edge router.
- PE2 removes the label from each packet and forwards the packets as IP packets.

This MPLS QoS enhancement allows service providers to classify packets according to their type, input interface, and other factors by setting (marking) each packet within the MPLS experimental field without changing the IP Precedence or DSCP field. For example, service providers can classify packets with or without considering the rate of the packets that PE1 receives. If the rate is a consideration, the service provider marks in-rate packets differently from out-of-rate packets.

Note The MPLS experimental bits allow you to specify the QoS for an MPLS packet. The IP Precedence/DSCP bits allow you to specify the QoS for an IP packet.

MPLS Infrastructure Changes Introduction of MFI and Removal of MPLS LSC and LC-ATM Features

This document explains the new MPLS Forwarding Infrastructure (MFI) and removal of support for MPLS label switch controller (LSC) and label-controlled ATM (LC-ATM) features and commands.

Finding Feature Information

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

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

Information About MPLS Infrastructure Changes

Introduction of the MPLS Forwarding Infrastructure

The MPLS control plane software is enhanced to make MPLS more scalable and flexible. The MFI, which manages MPLS data structures used for forwarding, replaces the Label Forwarding Information Base (LFIB).

Note The MFI and LFIB do not coexist in the same image. For a list of supported releases, see the "Feature Information for MPLS Forwarding Infrastructure."

Introduction of IP Rewrite Manager

Cisco software introduces a module called the MPLS IP Rewrite Manager (IPRM) that manages the interactions between Cisco Express Forwarding, the IP Label Distribution Modules (LDMs), and the MFI. MPLS IPRM is enabled by default. You need not configure or customize the IPRM. These commands are related to IPRM:

- **clear mpls ip iprm counters**
- **debug mpls ip iprm**
- **debug mpls ip iprm cef**
- **debug mpls ip iprm events**
- **debug mpls ip iprm ldm**
- **debug mpls ip iprm mfi**
- **show mpls ip iprm counters**
- **show mpls ip iprm ldm**

For information about these commands, see the *Cisco IOS Debug Command Reference* and the *Cisco IOS MPLS Command Reference*.

Removal of Support for MPLS LSC and LC-ATM Features

The following MPLS LSC and LC-ATM features are no longer supported, starting with Cisco IOS Release 12.4(20)T:

- MPLS LSC
- LC-ATM
- MPLS Scalability Enhancements for LSC and ATM LSR
- MPLS LSC Redundancy
- MPLS--OAM Insertion and Loop Detection on LC-ATM
- MPLS CoS Multi-VC Mode for PA-A3
- MPLS over ATM: Virtual Circuit Merge
- MPLS Diff-Serv Aware Traffic Engineering over ATM
- VSI Master MIB

MPLS LSC and LC-ATM Configurations

Before upgrading to Cisco IOS Release 12.4(20)T, remove all the MPLS LSC and LC-ATM configurations from the routers in your network. If your core network has ATM links, you can use packet-based MPLS. See the MPLS Label Distribution Protocol Overview for more information. If you provide ATM access to customers, you can use the Any Transport over MPLS: ATM over MPLS feature. See Any Transport over MPLS for more information.

If you have MPLS LSC or LC-ATM features configured and you upgrade to Cisco IOS Release 12.4(20)T, the configuration is not accepted. The system displays “unrecognized command” errors for any commands that are no longer supported.

Removal of Support for MPLS LSC and LC-ATM Commands

The following commands are no longer supported, starting with Cisco IOS Release 12.4(20)T:

- **debug mpls atm-cos**
- **debug mpls atm-ldp api**
- **debug mpls atm-ldp failure**
- **debug mpls atm-ldp routes**
- **debug mpls atm-ldp states**
- **debug mpls xmpls cross-connect**
- **debug mpls xmpls errors**
- **debug mpls xmpls events**
- **debug mpls xmpls vc**
- **debug mpls xtagatm cross-connect**
- **debug mpls xtagatm errors**
- **debug mpls xtagatm events**
- **debug mpls xtagatm vc**
- **debug vsi api**
- **debug vsi errors**
- **debug vsi events**
- **debug vsi packets**
- **debug vsi param-groups**
- **extended-port**
- **interface xtagatm**
- **mpls atm control-vc**
- **mpls atm cos**
- **mpls atm disable-headend-vc**
- **mpls atm multi-vc**
- **mpls atm vpi**
- **mpls atm vp-tunnel**
- **mpls cos-map**
- **mpls ldp atm control-mode**
- **mpls ldp atm vc-merge**s
- **mpls prefix-map**
- **mpls request-labels for**
- **mpls traffic-eng atm cos global-pool**
- **mpls traffic-eng atm cos sub-pool**
- **show controllers vsi control-interface**
- **show controllers vsi descriptor**
- **show controllers vsi session**
- **show controllers vsi status**
- **show controllers vsi traffic**
- **show controllers xmpls**
- **show controllers xtagatm**
- **show interface xtagatm**
- **show mpls atm-ldp bindings**

- **show mpls atm-ldp bindwait**
- **show mpls atm-ldp capability**
- **show mpls atm-ldp summary**
- **show mpls cos-map**
- **show mpls prefix-map**
- **show xtagatm cos-bandwidth-allocation**
- **show xtagatm cross-connect**
- **show xtagatm vc**
- **snmp-server enable traps vsimaster**
- **tag-control-protocol vsi**

Additional References

Related Documents

Related Topic	Document Title
MPLS commands	*Cisco IOS MPLS Command Reference*
MPLS Label Distribution Protocol	MPLS Label Distribution Protocol Overview
Layer 2 VPN features over MPLS	Any Transport over MPLS

Technical Assistance

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

Feature Information for MPLS Infrastructure Changes

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

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

Table 5 Feature Information for MPLS Infrastructure Changes

Feature Name	Releases	Feature Information
MPLS Infrastructure Changes	12.4(20)T Cisco IOS XE Release 3.5S	In Cisco IOS Release 12.4(20)T, this feature was introduced. In Cisco IOS XE Release 3.5S, support was added for the Cisco ASR 903 Router.

MPLS-Multilink PPP Support

The MPLS--Multilink PPP Support feature ensures that MPLS Layer 3 Virtual Private Networks (VPNs) with quality of service (QoS) can be enabled for bundled links. This feature supports Multiprotocol Label Switching (MPLS) over Multilink PPP (MLP) links in the edge (provider edge [PE]-to-customer edge [CE]) or in the MPLS core (PE-to-PE and PE-to-provider router [P]).

Service providers that use relatively low-speed links can use MLP to spread traffic across them in their MPLS networks. Link fragmentation and interleaving (LFI) should be deployed in the CE-to-PE link for efficiency, where traffic uses a lower link bandwidth (less than 768 kbps).

Finding Feature Information

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

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

Prerequisites for MPLS--Multilink PPP Support

The MPLS--Multilink PPP Support feature requires the following:

- Cisco Express Forwarding or distributed Cisco Express Forwarding enabled
- MPLS enabled on PE and P routers
- Cisco Express Forwarding switching enabled on the interface with the **ip route-cache cef**command

The first table below lists the required port adapters and processors for the MPLS--Multilink PPP Support feature on the Cisco 7200 series routers. The second table below lists the required port adapters and processors for the MPLS--Multilink PPP Support feature on the Cisco 7500 series routers.

Table 6 Required Cisco 7200 Port Adapters and Processors for MPLS--Multilink PPP Support

Port Adapter	Processor
PA-4T+ PA-8T Channelized adapters • PA-MC-2E1/120 • PA-MC-2T1 • PA-MC-2T3+ • PA-MC-4T1 • PA-MC-8E1/120 • PA-MC-8T1 • PA-MC-E3 • PA-MC-STM-1MM • PA-MC-STM-1SMI • PA-MC-T3 • PA-MC-8TE1+	Network processing engine models • NPE-400 • NPE-G1 • NSE-1

Table 7 Required Cisco 7500 Port Adapters and Processors for MPLS--Multilink PPP Support

Port Adapter	Processor
PA-4T+ PA-8T Channelized adapters • PA-MC-2E1/120 • PA-MC-2T1 • PA-MC-2T3+ • PA-MC-4T1 • PA-MC-8E1/120 • PA-MC-8T1 • PA-MC-E3 • PA-MC-STM-1MM • PA-MC-STM-1SMI • PA-MC-T3 • PA-MC-8TE1+	Route Switch Processors • RSP16 • RSP8 • RSP4+ Versatile interface processors • VIP4-50 • VIP4-80 • VIP6-80

Restrictions for MPLS--Multilink PPP Support

The MPLS--Multilink PPP Support feature is limited byplatform-specific restrictions that apply to the use of MLP and distributed MLP (dMLP).

For restrictions that apply to dMLP on the Cisco 7500 routers, see the Distributed Multilink Point-to-Point Protocol for Cisco 7500 Series Routers feature module.

Information About MPLS--Multilink PPP Support

MPLS Features Supported for Multilink PPP

The following topics provide information about MPLS features supported for MLP:

MPLS Layer 3 Virtual Private Network Features Supported for Multilink PPP

The table below lists MPLS Layer 3 VPN features supported for MLP and indicates if the feature is supported on CE-to-PE links, PE-to-P links, and Carrier Supporting Carrier (CSC) CE-to-PE links.

Table 8 MPLS Layer 3 VPN Features Supported for MLP

MPLS L3 VPN Feature	CE-to-PE Links	PE-to-P Links	CSC CE-to-PE Links
Static routes	Supported	--[1]	--
External Border Gateway Protocol (eBGP)	Supported	Not applicable to this configuration	Supported
Intermediate System-to-Intermediate System (IS-IS)	--	Supported	--
Open Shortest Path first (OSPF)	Supported	Supported	--
Enhanced Interior Gateway Routing Protocol (EIGRP)	Supported	Supported	--

1 An em dash (--) indicates that the configuration is not supported.

MPLS L3 VPN Feature	CE-to-PE Links	PE-to-P Links	CSC CE-to-PE Links
Interprovider (Inter-AS) VPNs (with Label Distribution Protocol [LDP])	Not applicable to this configuration	Supported (MLP between Autonomous System Border routers {ASBRs])	Not applicable to this configuration
Inter-AS VPNs with IPv4 Label Distribution	Not applicable to this configuration	Supported (MLP between ASBRs]	Not applicable to this configuration
CSC VPNs (with LDP)	--	Not applicable to this configuration	Supported
CSC VPNs with IPv4 label distribution	Supported	Not applicable to this configuration	Supported
External and internal BGP (eiBGP) Multipath	--	--	Not applicable to this configuration
Internal BGP (iBGP) Multipath	Not applicable to this configuration	--	Not applicable to this configuration
eBGP Multipath	--	--	--

MPLS Quality of Service Features Supported for Multilink PPP

The table below lists the MPLS QoS features supported for MLP and indicates if the feature is supported on CE-to-PE links, PE-to-P links, and CSC-CE-to-CSC-PE links.

Table 9 MPLS QoS Features Supported for MLP

MPLS QoS Feature	CE-to-PE Links	PE-to-P Links	CSC-CE-to-PE Links
Default copy of IP Precedence to EXP bits and the reverse	Supported	--[2]	--
Set MPLS EXP bits using the modular QoS Command-Line Interface (MQC)	Supported	Supported	Supported
Matching on MPLS EXP using MQC	Supported	Supported	Supported
Low Latency Queueing (LLQ)/ Class-Based Weighted Fair Queueing (CBWFQ) support	Supported	Supported	Supported

2 An em dash (--) indicates that the configuration is not supported.

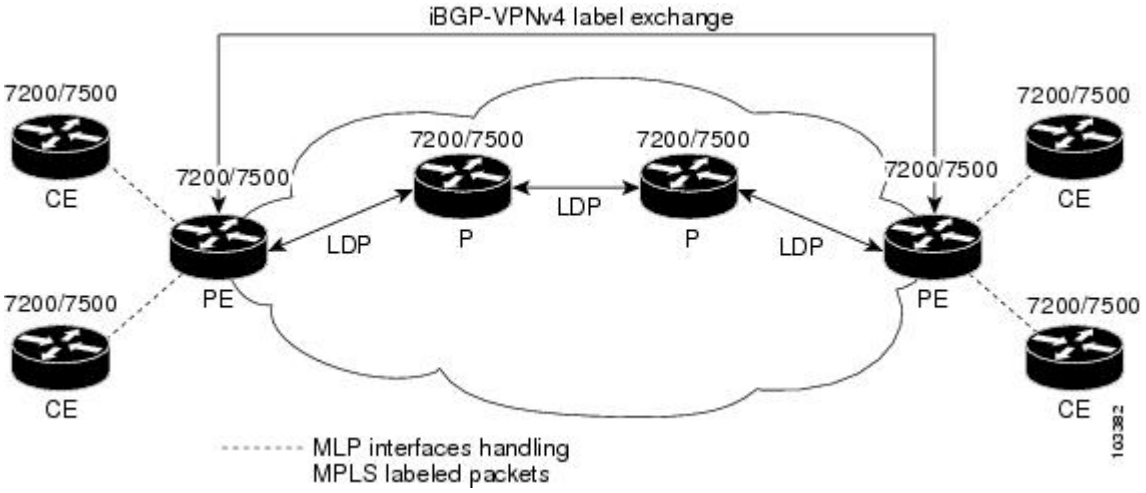
MPLS QoS Feature	CE-to-PE Links	PE-to-P Links	CSC-CE-to-PE Links
Weighted Random Early Detection (WRED) based on EXP bits using MQC	Supported	Supported	Supported
Policer with EXP bit-marking using MQC-3 action	Supported	Supported	Supported
Support for EXP bits in MPLS accounting	Supported	Supported	Supported

MPLS--Multilink PPP Support and PE-to-CE Links

The figure below shows a typical MPLS network in which the PE router is responsible for label imposition (at ingress) and disposition (at egress) of the MPLS traffic.

In this topology, MLP is deployed on the PE-to-CE links. The VPN routing and forwarding instance (VRF) interface is in a multilink bundle. There is no MPLS interaction with MLP; all packets coming into the MLP bundle are IP packets.

Figure 1 MLP and Traditional PE-to-CE Links



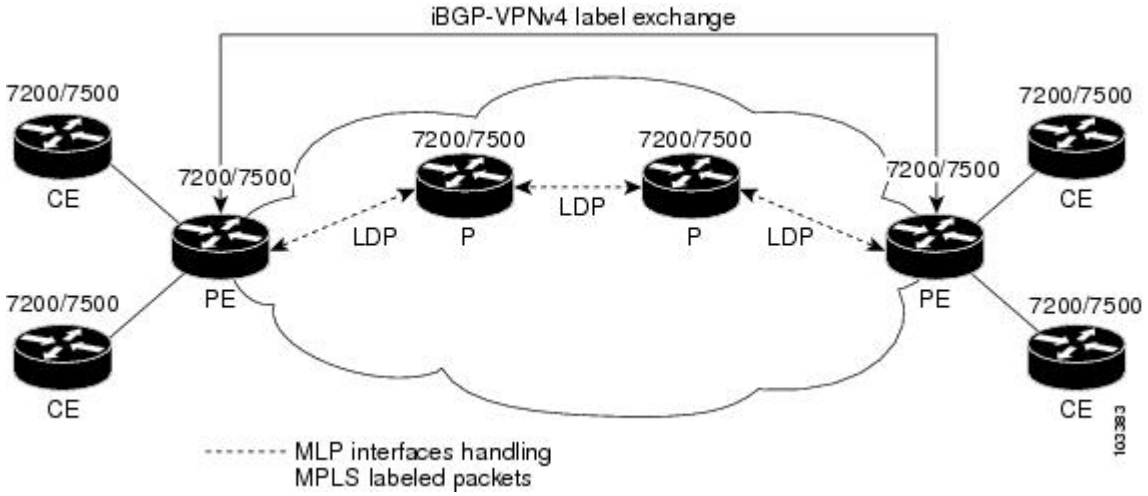
The PE-to-CE routing protocols that are supported for the MPLS--Multilink PPP Support feature are eBGP, OSPF, and EIGRP. Static routes are also supported between the CE and PE routers.

QoS features that are supported for the MPLS--Multilink PPP Support feature on CE-to-PE links are LFI, compressed Real-Time Transport Protocol (cRTP), policing, marking, and classification.

MPLS--Multilink PPP Support and Core Links

The figure below shows a sample topology in which MPLS is deployed over MLP on PE-to-P and P-to-P links. Enabling MPLS on MLP for PE-to-P links is similar to enabling MPLS on MLP for P-to-P links.

Figure 2 MLP on PE-to-P and P-to-P Links



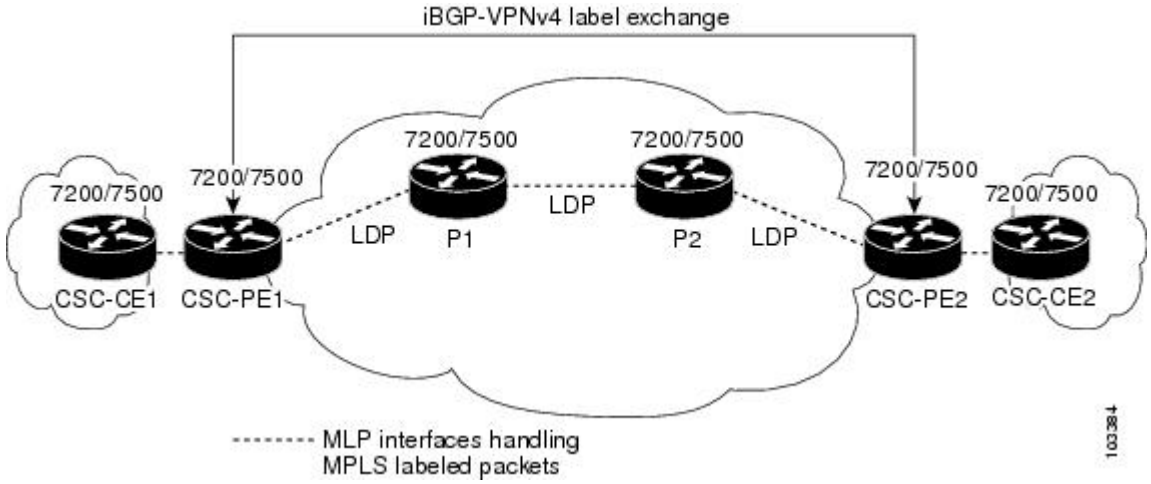
You employ MLP in the PE-to-P or P-to-P links primarily so that you can reduce the number of Interior Gateway Protocol (IGP) adjacencies and facilitate the load sharing of traffic.

In addition to requiring MLP on the PE-to-P links, the MPLS--Multilink PPP Support feature requires the configuration of an IGP routing protocol and LDP.

MPLS--Multilink PPP Support in a CSC Network

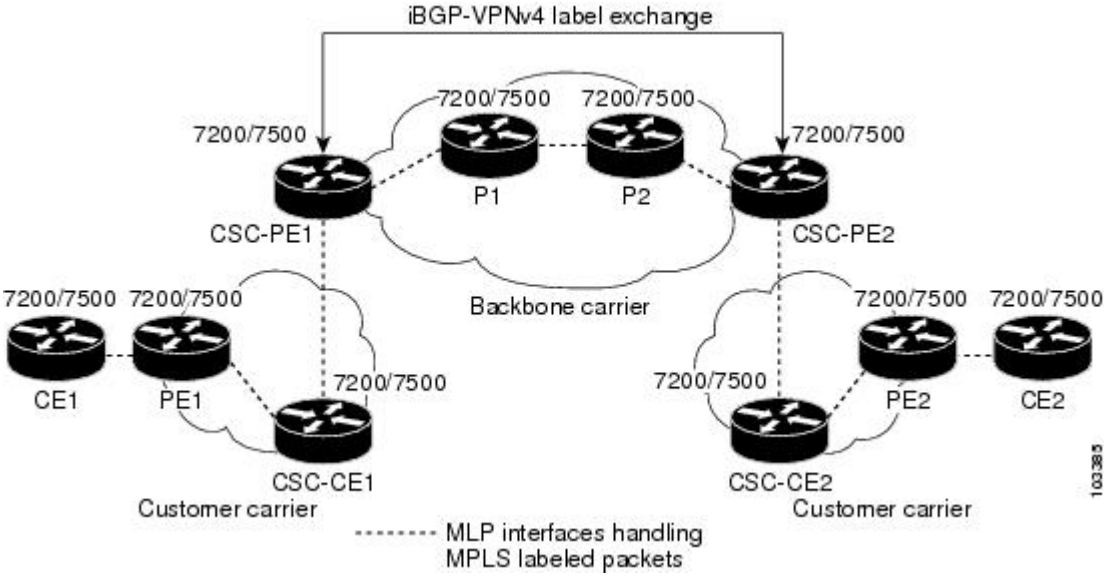
The figure below shows a typical MPLS VPN CSC network where MLP is configured on the CSC-CE-to-CSC-PE links.

Figure 3 MLP on CSC-CE-to-CSC-PE Links with MPLS VPN Carrier Supporting Carrier



The MPLS--Multilink PPP Support feature supports MLP between CSC-CE and CSC-PE links with LDP or with EBGP IPv4 label distribution. This feature also supports LFI for an MPLS VPN CSC configuration. The figure below shows all MLP links that this feature supports for CSC configurations.

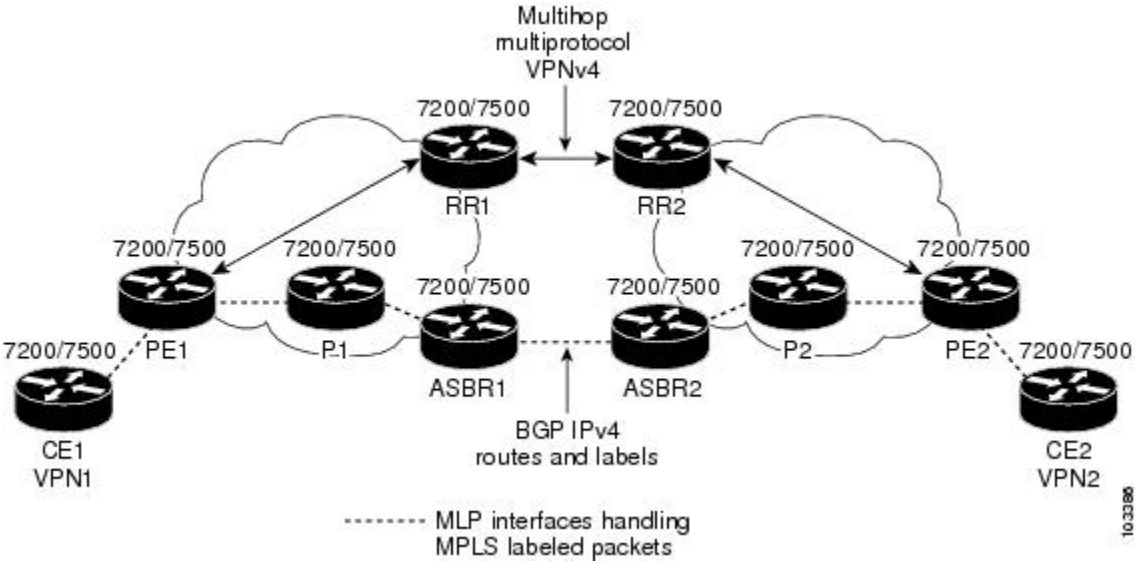
Figure 4 MLP Supported Links with MPLS VPN Carrier Supporting Carrier



MPLS--Multilink PPP Support in an Interautonomous System

The figure below shows a typical MPLS VPN interautonomous system (Inter-AS) network where MLP is configured on the PE-to-CE links.

Figure 5 MLP on ASBR-to-PE Links in an MPLS VPN Inter-AS Network



The MPLS--Multilink PPP Support feature supports MLP between ASBR links for Inter-AS VPNs with LDP and with eBGP IPv4 label distribution.

How to Configure MPLS--Multilink PPP Support

Service providers that use relatively low-speed links can use MLP to spread traffic across them in their MPLS networks. LFI should be deployed in the CE-to-PE link for efficiency, where traffic uses lower link bandwidth (less than 768 kbps). The MPLS--Multilink PPP Support feature can reduce the number of IGP adjacencies and facilitate load sharing of traffic.

The tasks in this section can be performed on CE-to-PE links, PE-to-P links, P-to-P links, and CSC CSC-CE-to-CSC-PE links.

Enabling Cisco Express Forwarding or Distributed Cisco Express Forwarding Switching

Perform the following task to enable Cisco Express Forwarding or distributed Cisco Express Forwarding switching.

Multilink PPP requires the configuration of standard Cisco Express Forwarding. Distributed MLP (dMLP) requires the configuration of distributed Cisco Express Forwarding.

Cisco Express Forwarding is enabled by default on most Cisco platforms running Cisco IOS software Release12.0 or a later release. To find out if Cisco Express Forwarding is enabled on your platform, enter the**showipcef**command. If Cisco Express Forwarding is enabled, you receive output that looks like this:

```
Router# show ip cef
Prefix              Next Hop            Interface
10.2.61.8/24        192.168.100.1       FastEthernet1/0/0
                    192.168.101.1       FastEthernet6/1
```

If Cisco Express Forwarding is not enabled on your platform, the output for the **showipcef**command looks like this:

```
Router# show ip cef
%CEF not running
```

Distributed Cisco Express Forwarding is enabled by default on devices such as the Catalyst 6500 series switch, the Cisco 7500 series router, and the Cisco 12000 series Internet router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Do one of the following:
   - **ip cef**
   -
   - **ip cef distributed**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	Do one of the following:
	• **ip cef**
	•
	• **ip cef distributed**
	
	Example:
	`Router(config)# ip cef`
	
	Example:
	
	Example:
	`Router(config)# ip cef distributed`
		Enables standard Cisco Express Forwarding switching.
		or
		Enables distributed Cisco Express Forwarding switching.

	Command or Action	Purpose
Step 4	**exit** **Example:** `Router(config)# exit`	Exits to privileged EXEC mode.

Creating a Multilink Bundle

Perform this task to create a multilink bundle for the MPLS--Multilink PPP Support feature. This can reduce the number of IGP adjacencies and facilitate load sharing of traffic.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface multilink** *group-number*
4. **ip address** *address mask* [**secondary**]
5. **encapsulation** *encapsulation-type*
6. **ppp multilink**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface multilink** *group-number* **Example:** `Router(config)# interface multilink 1`	Creates a multilink bundle or enters multilink interface configuration mode. • The *group-number* argument is the number of the multilink bundle (a nonzero number).

	Command or Action	Purpose
Step 4	**ip address** *address mask* [**secondary**] **Example:** `Router(config-if)# ip address` *address mask*	Sets a primary or secondary IP address for an interface. • The *address* argument is the IP address. • The *mask* argument is the mask for the associated IP subnet. • The **secondary** keyword specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address. This command is used to assign an IP address to the multilink interface.
Step 5	**encapsulation** *encapsulation-type* **Example:** `Router(config-if)# encapsulation ppp`	Sets the encapsulation method used by the interface. • The *encapsulation-type* argument specifies the encapsulation type. The keyword **ppp** enables PPP encapsulation.
Step 6	**ppp multilink** **Example:** `Router(config-if)# ppp multilink`	Enables MLP on an interface.
Step 7	**end** **Example:** `Router(config-if)#` **end**	Exits to privileged EXEC mode.

Assigning an Interface to a Multilink Bundle

Perform this task to assign an interface to a multilink bundle for the MPLS--Multilink PPP Support feature.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **controller** {**t1** | **e1**} *slot* / *port*
4. **channel-group** *channel-number* **timeslots** *range*
5. **exit**
6. **interface serial** *slot* / *port* **:** *channel-group*
7. **ip route-cache** [**cef** | **distributed**]
8. **no ip address**
9. **keepalive** [*period* [*retries*]]
10. **encapsulation** encapsulation-type
11. **multilink-group** *group-number*
12. **ppp multilink**
13. **ppp authentication chap**
14. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**controller** {**t1**	**e1**} *slot* / *port* **Example:** `Router# controller t1 1/3`	Configures a T1 or E1 controller and enters controller configuration mode. • The **t1**keyword indicates a T1 line card. • The **e1** keyword indicates an E1 line card. • The *slot* / *port*arguments are the backplane slot number and port number on the interface. Refer to your hardware installation manual for the specific slot numbers and port numbers.

	Command or Action	Purpose
Step 4	**channel-group** *channel-number* **timeslots** *range* **Example:** `Router(config-controller)# channel-group 1 timeslots 1`	Defines the time slots that belong to each T1 or E1 circuit. • The *channel-number* argument is the channel-group number. When a T1 data line is configured, channel-group numbers can be values from 0 to 23. When an E1 data line is configured, channel-group numbers can be values from 0 to 30. • The **timeslots** *range* keyword-argument pair specifies 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. You can specify a time slot range (for example, 1-29), individual time slots separated by commas (for example 1, 3, 5), or a combination of the two (for example 1-14, 15, 17-31).
Step 5	**exit** **Example:** `Router(config-controller)# exit`	Exits to global configuration mode.
Step 6	**interface serial** *slot* / *port* **:** *channel-group* **Example:** `Router(config)# interface serial 1/0:1`	Configures a serial interface for a Cisco 7500 series router with channelized T1 or E1 and enters interface configuration mode. • The *slot* argument indicates the slot number. Refer to the appropriate hardware manual for slot and port information. • The */port* argument indicates the port number. Refer to the appropriate hardware manual for slot and port information. • The *:channel-group* argument indicates the channel group number. Cisco 7500 series routers specify the channel group number in the range of 0 to 4 defined with the **channel-group**controller configuration command.
Step 7	**ip route-cache** [**cef** | **distributed**] **Example:** `Router(confg-if)# ip route-cache cef`	Controls the use of switching methods for forwarding IP packets • The **cef** keyword enables Cisco Express Forwarding operation on an interface after Cisco Express Forwarding operation was disabled. • The **distributed** keyword enables distributed switching on the interface.
Step 8	**no ip address** **Example:** `Router(config-if)# no ip address`	Removes any specified IP address.

	Command or Action	Purpose
Step 9	**keepalive** [*period* [*retries*]] **Example:** `Router(config-if)# keepalive`	Enables keepalive packets and specifies the number of times that the Cisco IOS software tries to send keepalive packets without a response before bringing down the interface or before bringing the tunnel protocol down for a specific interface. • The *period* argument is an integer value, in seconds, greater than 0. The default is 10. • The *retries*argument specifies the number of times that the device will continue to send keepalive packets without response before bringing the interface down. Enter an integer value greater than 1 and less than 255. If you do not enter a value, the value that was previously set is used; if no value was specified previously, the default of 5 is used. If you are using this command with a tunnel interface, the command specifies the number of times that the device will continue to send keepalive packets without response before bringing the tunnel interface protocol down.
Step 10	**encapsulation** encapsulation-type **Example:** `Router(config-if)# encapsulation ppp`	Sets the encapsulation method used by the interface. • The *encapsulation-type* argument specifies the encapsulation type. The keyword **ppp** enables PPP encapsulation.
Step 11	**multilink-group** *group-number* **Example:** `Router(config-if)# multilink-group 1`	Designates an interface as part of a multilink leased line bundle. • The *group-number* argument is the number of the multilink bundle (a nonzero number).
Step 12	**ppp multilink** **Example:** `Router(config-if)# ppp multilink`	Enables MLP on an interface.
Step 13	**ppp authentication chap** **Example:** `Router(config-if)# ppp authentication chap`	(Optional) Enables Challenge Handshake Authentication Protocol (CHAP) authentication on a serial interface.

	Command or Action	Purpose
Step 14	**end** **Example:** `Router(config-if)# end`	Exits to privileged EXEC mode.

Disabling PPP Multilink Fragmentation

Perform this task to disable PPP multilink fragmentation. PPP multilink fragmentation is enabled by default.

Enabling fragmentation reduces the delay latency among bundle links, but adds some load to the CPU. Disabling fragmentation might produce better throughput.

If your data traffic is consistently of a similar size, we recommend disabling fragmentation. In this case, the benefits of fragmentation can be outweighed by the added load on the CPU.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ppp multilink fragmentation disable**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** *type number* **Example:** `Router(config)# interface serial 1/0/0`	Configures an interface type and enters interface configuration mode. • The *type* argument indicates the type of interface to be configured. • The *number* argument specifies the port, connector, or interface card number. The numbers are assigned at the factory at the time of installation or when the interface is added to a system, and can be displayed with the **show interfaces** command.
Step 4	**ppp multilink fragmentation disable** **Example:** `Router(config-if)# ppp multilink fragmentation disable`	Disables packet fragmentation.
Step 5	**end** **Example:** `Router(config-if)# end`	Exits to privileged EXEC mode.

Verifying the Multilink PPP Configuration

SUMMARY STEPS

1. **enable**
2. **show ip interface brief**
3. **show ppp multilink**
4. **show ppp multilink interface** *interface-bundle*
5. **show interface** *interface-name interface-number*
6. **show mpls forwarding-table**
7. **exit**

DETAILED STEPS

Step 1 **enable**

Use this command to enable privileged EXEC mode. Enter your password if prompted. For example:

Example:

```
Router> enable
Router#
```

Step 2 **show ip interface brief**

Use this command to verify logical and physical MLP interfaces. For example:

Example:

```
Router# show ip interface brief
Locolrface                  IP-Address      OK? Method Status                Prot
FastEthernet1/0/0         10.3.62.106      YES NVRAM  up                    up
FastEthernet0/0/1          unassigned      YES NVRAM  administratively down down
FastEthernet0/0/0          unassigned      YES NVRAM  administratively down down
FastEthernet0/0/1          unassigned      YES NVRAM  administratively down down
FastEthernet0/0/2          unassigned      YES NVRAM  administratively down down
FastEthernet0/1/0          unassigned      YES NVRAM  administratively down down
FastEthernet0/1/1          unassigned      YES NVRAM  administratively down down
FastEthernet0/1/2          unassigned      YES NVRAM  administratively down down
FastEthernet1/2/0          unassigned      YES NVRAM  administratively down down
FastEthernet1/0/1          unassigned      YES NVRAM  administratively down down
FastEthernet1/1/0          unassigned      YES NVRAM  administratively down down
FastEthernet1/1/1          unassigned      YES NVRAM  administratively down down
FastEthernet1/1/2          unassigned      YES NVRAM  administratively down down
Serial1/1/0:1              unassigned      YES NVRAM  administratively down down
Serial1/1/0:2              unassigned      YES NVRAM  administratively down down
Serial1/1/1:1              unassigned      YES NVRAM  up                    up
Serial1/1/1:2              unassigned      YES NVRAM  up                    down
Serial1/1/3:1              unassigned      YES NVRAM  up                    up
Serial1/1/3:2              unassigned      YES NVRAM  up                    up
Multilink6                 10.30.0.2       YES NVRAM  up                    up
Multilink8                 unassigned      YES NVRAM  administratively down down
Multilink10                10.34.0.2       YES NVRAM  up                    up
Loopback0                  10.0.0.1        YES NVRAM  up                    up
```

Step 3 **show ppp multilink**

Use this command to verify that you have created a multilink bundle. For example:

Example:

```
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
```

Step 4 **show ppp multilink interface** *interface-bundle*

Use this command to display information about a specific MLP interface. For example:

Example:

```
Router# show ppp multilink interface multilink6
Multilink6, bundle name is router
  Bundle up for 00:42:46, 1/255 load
  Receive buffer limit 24384 bytes, frag timeout 1524 ms
  Bundle is Distributed
    0/0 fragments/bytes in reassembly list
    1 lost fragments, 48 reordered
    0/0 discarded fragments/bytes, 0 lost received
    0x4D7 received sequence, 0x0 sent sequence
  Member links: 2 active, 0 inactive (max not set, min not set)
    Se1/1/3:1, since 00:42:46, 240 weight, 232 frag size
    Se1/1/3:2, since 00:42:46, 240 weight, 232 frag size
```

Step 5 **show interface** *interface-name interface-number*

Use this command to display information about serial interfaces in your configuration. For example:

Example:

```
Router# show interface serial 1/1/3:1
Serial1/1/3:1 is up, line protocol is up
  Hardware is Multichannel T1
  MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open, multilink Open, crc 16, Data non-inverted
  Last input 00:00:01, output 00:00:01, output hang never
  Last clearing of "show interface" counters 00:47:13
  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
     722 packets input, 54323 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
     697 packets output, 51888 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions no alarm present
  Timeslot(s) Used:1, subrate: 64Kb/s, transmit delay is 0 flags
  Transmit queue length 25
Router# show interface serial 1/1/3:2
Serial1/1/3:2 is up, line protocol is up
  Hardware is Multichannel T1
  MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open, multilink Open, crc 16, Data non-inverted
  Last input 00:00:03, output 00:00:03, output hang never
  Last clearing of "show interface" counters 00:47:16
  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
     725 packets input, 54618 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
     693 packets output, 53180 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions no alarm present
  Timeslot(s) Used:2, subrate: 64Kb/s, transmit delay is 0 flags
  Transmit queue length 26
```

You can also use the **show interface** command to display information about the multilink interface:

Example:

```
Router# show interface multilink6
Multilink6 is up, line protocol is up
  Hardware is multilink group interface
  Internet address is 10.30.0.2/8
  MTU 1500 bytes, BW 128 Kbit, DLY 100000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open, multilink Open
  Open: CDPCP, IPCP, TAGCP, loopback not set
  DTR is pulsed for 2 seconds on reset
  Last input 00:00:00, output never, output hang never
  Last clearing of "show interface" counters 00:48:43
  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
     1340 packets input, 102245 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
     1283 packets output, 101350 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

Step 6 **show mpls forwarding-table**

Use this command to display contents of the MPLS Label Forwarding Information Base (LFIB) and look for information on multilink interfaces associated with a point2point next hop. For example:

Example:

```
Router# show mpls forwarding-table
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
16     Untagged    10.30.0.1/32      0          Mu6        point2point
17     Pop tag     10.0.0.3/32       0          Mu6        point2point
18     Untagged    10.0.0.9/32[V]    0          Mu10       point2point
19     Untagged    10.0.0.11/32[V]   6890       Mu10       point2point
20     Untagged    10.32.0.0/8[V]    530        Mu10       point2point
21     Aggregate   10.34.0.0/8[V]    0
22     Untagged    10.34.0.1/32[V]   0          Mu10       point2point
```

Use the **show ip bgp vpnv4**command to display VPN address information from the Border Gateway Protocol (BGP) table:

Example:

```
Router# show ip bgp vpnv4 all summary
BGP router identifier 10.0.0.1, local AS number 100
BGP table version is 21, main routing table version 21
10 network entries using 1210 bytes of memory
10 path entries using 640 bytes of memory
2 BGP path attribute entries using 120 bytes of memory
1 BGP extended community entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1994 total bytes of memory
BGP activity 10/0 prefixes, 10/0 paths, scan interval 5 secs
10.0.0.3  4 100 MsgRc52 MsgSe52  TblV21  0  0 00:46:35 State/P5xRcd
```

Step 7 **exit**

Use this command to exit to user EXEC mode. For example:

Example:

```
Router# exit
Router>
```

Configuration Examples for MPLS--Multilink PPP Support

Sample MPLS--Multilink PPP Support Configurations

The following examples show sample configurations for MLP on a Cisco 7200 router, on a Cisco 7500 router, and on a CSC network. The configuration of MLP on an interface is the same for PE-to-CE links, PE-to-P links, and P-to-P links.

Sample Multilink PPP Configuration on Cisco 7200 Series Router

Following is a sample configuration of a Cisco 7200 router, which is connected with a T1 line card and configured with an MPLS Multilink PPP interface:

```
controller T1 1/3
 framing esf
 clock source internal
 linecode b8zs
 channel-group 1 timeslots 1
 channel-group 2 timeslots 2
 no yellow generation
 no yellow detection
!
interface Multilink6
 ip address 10.37.0.1 255.0.0.0
 ppp multilink interleave
 tag-switching ip
 load-interval 30
 multilink-group 6
!
interface Serial1/3:1
 encapsulation ppp
 no ip address
 ppp multilink
 tx-queue-limit 26
 multilink-group 6
 peer neighbor-route
!
interface Serial1/3:2
 encapsulation ppp
 no ip address
 ppp multilink
 tx-queue-limit 26
 multilink-group 6
 peer neighbor-route
```

Sample Multilink PPP Configuration for Cisco 7500 Series Router

Following is a sample configuration of a Cisco 7500 router, which is connected with a T1 line card and configured with an MPLS Multilink PPP interface:

```
controller T1 1/1/3
 framing esf
 clock source internal
 linecode b8zs
 channel-group 1 timeslots 1
```

```
 channel-group 2 timeslots 2
 no yellow generation
 no yellow detection
!
interface Multilink6
 ip address 10.37.0.2 255.0.0.0
 ppp multilink interleave
 tag-switching ip
 load-interval 30
 multilink-group 6
!
interface Serial1/1/3:1
 encapsulation ppp
 no ip address
 ppp multilink
 tx-queue-limit 26
 multilink-group 6
 peer neighbor-route
!
interface Serial1/1/3:2
 encapsulation ppp
 no ip address
 ppp multilink
 tx-queue-limit 26
 multilink-group 6
 peer neighbor-route
```

Sample Multilink PPP Configuration on an MPLS CSC PE Router

Following is a sample configuration for an MPLS CSC PE router. An EBGP session is configured between the PE and CE routers.

```
PE-Router# show running-config interface Serial1/0:1
Building configuration...
!
mpls label protocol ldp
ip cef
ip vrf vpn2
 rd 200:1
 route-target export 200:1
 route-target import 200:1
!
controller T1 1/0
 framing esf
 clock source internal
 linecode b8zs
 channel-group 1 timeslots 1
 channel-group 2 timeslots 2
 no yellow generation
 no yellow detection
!
interface Serial1/0:1
 no ip address
 encapsulation ppp
 tx-ring-limit 26
 ppp multilink
 ppp multilink group 1
!
interface Serial1/0:2
 no ip address
 encapsulation ppp
 tx-ring-limit 26
 ppp multilink
 ppp multilink group 1
!
interface Multilink1
 ip vrf forwarding vpn2
 ip address 10.35.0.2 255.0.0.0
 no peer neighbor-route
 load-interval 30
 ppp multilink
```

```
 ppp multilink interleave
 ppp multilink group 1
!
!
router ospf 200
 log-adjacency-changes
 auto-cost reference-bandwidth 1000
 redistribute connected subnets
 passive-interface Multilink1
 network 10.0.0.7 0.0.0.0 area 200
 network 10.31.0.0 0.255.255.255 area 200
!
!
 router bgp 200
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 neighbor 10.0.0.11 remote-as 200
 neighbor 10.0.0.11 update-source Loopback0
 !
 address-family vpnv4
 neighbor 10.0.0.11 activate
 neighbor 10.0.0.11 send-community extended
 bgp scan-time import 5
 exit-address-family
 !
 address-family ipv4 vrf vpn2
 redistribute connected
 neighbor 10.35.0.1 remote-as 300
 neighbor 10.35.0.1 activate
 neighbor 10.35.0.1 as-override
 neighbor 10.35.0.1 advertisement-interval 5
 no auto-summary
 no synchronization
 exit-address-family
```

Enabling Cisco Express Forwarding or Distributed Cisco Express Forwarding Example

The following example shows how to enable Cisco Express Forwarding for MLP configurations:

```
enable
configure terminal
ip cef
```

The following example shows how to enable distributed Cisco Express Forwarding for dMLP configurations:

```
enable
configure terminal
ip cef distribute
```

Creating a Multilink Bundle Example

The following example shows how to create a multilink bundle for the MPLS--Multilink PPP Support feature:

```
interface multilink 1
 ip address 10.0.0.0 10.255.255.255
 encapsulation ppp
 ppp chap hostname group 1
 ppp multilink
 multilink-group 1
```

Assigning an Interface to a Multilink Bundle Example

The following example shows how to create four multilink interfaces with distributed Cisco Express Forwarding switching and MLP enabled. Each of the newly created interfaces is added to a multilink bundle.

```
interface multilink1
 ip address 10.0.0.0 10.255.255.255
 ppp chap hostname group 1
 ppp multilink
 multilink-group 1

interface serial 1/0/0/:1
 no ip address
 encapsulation ppp
 ip route-cache distributed
 no keepalive
 ppp multilink
 multilink-group 1
interface serial 1/0/0/:2
 no ip address
 encapsulation ppp
 ip route-cache distributed
 no keepalive
 ppp chap hostname group 1
 ppp multilink
 multilink-group 1
interface serial 1/0/0/:3
 no ip address
 encapsulation ppp
 ip route-cache distributed
 no keepalive
 ppp chap hostname group 1
 ppp multilink
 multilink-group 1
interface serial 1/0/0/:4
 no ip address
 encapsulation ppp
 ip route-cache distributed
 no keepalive
 ppp chap hostname group 1
 ppp multilink
 multilink-group
```

Additional References

The following sections provide references related to the MPLS--Multilink PPP Support feature:

Related Documents

Related Topic	Document Title
Configuration tasks for Distributed MLP for Cisco 7500 series routers	Distributed Multilink Point-to-Point Protocol for Cisco 7500 Series Routers
Configuration tasks for media-independent PPP and Multilink PPP	Configuring Media-Independent PPP and Multilink PPP
Configuration tasks for MPLS DiffServ tunneling modes	MPLS DiffServ Tunneling Modes

Related Topic	Document Title
Configuration tasks for the MPLS QoS multi-VC mode feature	"Configuring MPLS" chapter, Cisco IOS Multiprotocol Label Switching Configuration Guide , Release 12.4
Configuration tasks for MPLS VPNs	"MPLS Virtual Private Networks" chapter, Cisco IOS Multiprotocol Label Switching Configuration Guid e, Release 12.4
Configuration tasks for MPLS VPN CSC	"MPLS Virtual Private Networks" chapter, Cisco IOS Multiprotocol Label Switching Configuration Guid e, Release 12.4
Configuration tasks for MPLS VPN CSC with IPv4 BGP label distribution	"MPLS Virtual Private Networks" chapter, Cisco IOS Multiprotocol Label Switching Configuration Guid e, Release 12.4
Configuration tasks for MPLS VPN Inter-AS with IPv4 BGP label distribution	"MPLS Virtual Private Networks" chapter, Cisco IOS Multiprotocol Label Switching Configuration Guid e, Release 12.4

Standards

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

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 1990	*The PPP Multilink Protocol (MP)*

Technical Assistance

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

Command Reference

This feature uses no new or modified commands.

Feature Information for MPLS--Multilink PPP Support

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

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

Table 10 Feature Information for MPLS--Multilink PPP Support

Feature Name	Releases	Feature Information
MPLS--Multilink PPP Support	12.2(8)T 12.2(15)T1012.3(5a) 12.3(7)T 12.2(28)SB 12.4(20)T	The MPLS--Multilink PPP Support feature ensures that MPLS Layer 3 Virtual Private Networks (VPNs) with quality of service (QoS) can be enabled for bundled links. This feature supports Multiprotocol Label Switching (MPLS) over Multilink PPP (MLP) links in the edge (provider edge [PE]-to-customer edge [CE]) or in the MPLS core (PE-to-PE and PE-to-provider router [P]). Service providers that use relatively low-speed links can use MLP to spread traffic across them in their MPLS networks. Link fragmentation and interleaving (LFI) should be deployed in the CE-to-PE link for efficiency, where traffic uses a lower link bandwidth (less than 768 kbps). In 12.2(8)T, MLP support on CE-to-PE links was introduced. In 12.2(15)T10 and 12.3(5a), MLP support for MPLS networks was extended to PE-to-P links, PE-to-PE links, Carrier Supporting Carrier (CSC) CSC-CE-to-CSC-PE links, and interautonomous system (Inter-AS) PE-to-PE links. In 12.3(7)T, the feature was integrated into the Cisco IOS 12.3T release. In 12.2(28)SB, the feature was integrated into the Cisco IOS 12.2SB release. In 12.4(20)T, the feature was integrated into the Cisco IOS 12.4T release.

Glossary

bundle --A group of interfaces connected by parallel links between two systems that have agreed to use Multilink PPP (MLP) over those links.

CBWFQ --class-based weighted fair queueing. A queueing option that extends the standard Weighted Fair Queueing (WFQ) functionality to provide support for user-defined traffic classes.

Cisco Express Forwarding --A proprietary form of switching that optimizes network performance and scalability for networks with large and dynamic traffic patterns, such as the Internet, and for networks characterized by intensive web-based applications or interactive sessions. Although you can use Cisco Express Forwarding in any part of a network, it is designed for high-performance, highly resilient Layer 3 IP backbone switching.

EIGRP --Enhanced Interior Gateway Routing Protocol. An advanced version of the Interior Gateway Routing Protocol (IGRP) developed by Cisco. It provides superior convergence properties and operating efficiency, and combines the advantages of link-state protocols with those of distance vector protocols.

IGP --Interior Gateway Protocol. An Internet protocol used to exchange routing information within an autonomous system. Examples of common Internet IGPs include Interior Gateway Routing Protocol (IGRP), Open Shortest Path First (OSPF), and Routing Information Protocol (RIP).

IGRP --Interior Gateway Routing Protocol. An Interior Gateway Protocol (IGP) developed by Cisco to address the issues associated with routing in large, heterogeneous networks. Compare with Enhanced Interior Gateway Routing Protocol (EIGRP).

IS-IS --Intermediate System-to-Intermediate System. An Open Systems Interconnection (OSI) link-state hierarchical routing protocol, based on DECnet Phase V routing, in which IS-IS routers exchange routing information based on a single metric to determine network topology.

LCP --Link Control Protocol. A protocol that establishes, configures, and tests data link connections for use by PPP.

LFI --link fragmentation and interleaving. The Cisco IOS XE LFI feature reduces delay on slower-speed links by breaking up large datagrams and interleaving low-delay traffic packets with the smaller packets resulting from the fragmented datagram. LFI allows reserve queues to be set up so that Real-Time Protocol (RTP) streams can be mapped into a higher priority queue in the configured weighted fair queue set.

link --One of the interfaces in a bundle.

LLQ --low latency queueing. A quality of service QoS queueing feature that provides a strict priority queue (PQ) for voice traffic and weighted fair queues for other classes of traffic. It is also called priority queueing/class-based weighted fair queueing (PQ/CBWFQ).

MLP --Multilink PPP. A method of splitting, recombining, and sequencing datagrams across multiple logical links. The use of MLP increases throughput between two sites by grouping interfaces and then load balancing packets over the grouped interfaces (called a bundle). Splitting packets at one end, sending them over the bundled interfaces, and recombining them at the other end achieves load balancing.

MQC --Modular QoS CLI. MQC is a CLI structure that allows users to create traffic polices and attach these polices to interfaces. MQC allows users to specify a traffic class independently of QoS policies.

NCP --Network Control Protocol. A series of protocols for establishing and configuring different network layer protocols (such as for AppleTalk) over PPP.

OSPF --Open Shortest Path First. A link-state, hierarchical Interior Gateway Protocol (IGP) routing algorithm proposed as a successor to Routing Information Protocol (RIP) in the Internet community. OSPF features include least-cost routing, multipath routing, and load balancing. OSPF was derived from an early version of the IS-IS protocol.

PPP --Point-to-Point Protocol. A successor to the Serial Line Interface Protocol (SLIP) that provides router-to-router and host-to-network connections over synchronous and asynchronous circuits. PPP works with several network layer protocols (such as IP, Internetwork Packet Exchange [IPX], and AppleTalk Remote Access [ARA]). PPP also has built-in security mechanisms (such as Challenge Handshake Authentication Protocol [CHAP] and Password Authentication Protocol [PAP]). PPP relies on two protocols: Link Control Protocol (LCP) and Network Control Protocol (NCP).

RIP --Routing Information Protocol. A version of Interior Gateway Protocol (IGP) that is supplied with UNIX Berkeley Standard Distribution (BSD) systems. Routing Information Protocol (RIP) is the most common IGP in the Internet. It uses hop count as a routing metric.

Virtual Bundle Interface --An interface that represents the master link of a bundle. It is not tied to any physical interface. Data going over the bundle is transmitted and received through the master link.

WFQ --weighted fair queueing. A congestion management algorithm that identifies conversations (in the form of traffic streams), separates packets that belong to each conversation, and ensures that capacity is shared fairly among the individual conversations. WFQ is an automatic way of stabilizing network behavior during congestion and results in improved performance and reduced retransmission.

WRED --weighted random early detection. A queueing method that ensures that high-precedence traffic has lower loss rates than other traffic during times of congestion.

MPLS MTU Command Changes

This document explains the change in the behavior of the **mplsmtu** command for the following Cisco IOS releases:

- 12.2(27)SBC and later
- 12.2(33)SRA and later
- 12.2(33)SXH and later
- 12.4(11)T and later
- 15.0(1)M1
- 15.1(2)S

You cannot set the Multiprotocol Label Switching (MPLS) maximum transmission unit (MTU) to a value larger than the interface MTU value. This eliminates problems such as dropped packets, data corruption, and high CPU rates from occurring when the MPLS MTU value settings are larger than the interface MTU values. Cisco IOS software allows the MPLS MTU value to be higher than the interface MTU value only for interfaces that have a default interface MTU value of 1580 or less.

Note In Cisco IOS Release 15.0(1)M1, you can configure the interface MTU on PA-1FE and PA-2FE port adapters (PAs). The range of values is from 1500 to 1530. Before this enhancement, the MTU of those interfaces was not configurable, and any attempt to configure the interface MTU displayed the following message: *%Interface{InterfaceName}doesnotsupportusersettablemtu.*

Finding Feature Information

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

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

Information About MPLS MTU Command Changes

MPLS MTU Values During Upgrade

If you have configuration files with MPLS MTU values that are larger than the interface MTU values and you upgrade to Cisco IOS Release 12.2(27)SBC, 12.2(33)SRA, 12.4(11)T, 12.2(33)SXH, or later releases, the software does not change the MPLS MTU value. When you reboot the router, the software accepts the values that are set for the MPLS MTU and the interface MTU. The following error message is displayed during system initialization:

```
Setting the mpls mtu to xxxx on interface x/x, which is higher than the interface MTU
xxxx. This could lead to packet forwarding problems including packet drops.
You must set the MPLS MTU values equal to or lower than the interface MTU values.
```

Caution If you do not set the MPLS MTU less than or equal to the interface MTU, data corruption, dropped packets, and high CPU conditions can occur.

Guidelines for Setting MPLS MTU and Interface MTU Values

When configuring the network to use MPLS, set the core-facing interface MTU values greater than the edge-facing interface MTU values using one of the following methods:

- Set the interface MTU values on the core-facing interfaces to a higher value than the interface MTU values on the customer-facing interfaces to accommodate any packet labels, such as MPLS labels, that an interface might encounter. Make sure that the interface MTUs on the remote end interfaces have the same interface MTU values. The interface MTU values on both ends of the link must match.
- Set the interface MTU values on the customer-facing interfaces to a lower value than the interface MTU on the core-facing interfaces to accommodate any packet labels, such as MPLS labels, that an interface might encounter. When you set the interface MTU on the edge interfaces, ensure that the interface MTUs on the remote end interfaces have the same values. The interface MTU values on both ends of the link must match.

Changing the interface MTU can also modify the IP MTU, Connectionless Network Service (CLNS) MTU, and other MTU values because they depend on the value of the interface MTU. The Open Shortest Path First (OSPF) routing protocol requires that the IP MTU values match on both ends of the link. Similarly, the Intermediate System-to-Intermediate System (IS-IS) routing protocol requires that the CLNS MTU values match on both ends of the link. If the values on both ends of the link do not match, IS-IS or OSPF cannot complete initialization.

If the configuration of the adjacent router does not include the **mplsmtu**and **mtu** commands, add these commands to the router.

Note The MPLS MTU setting is displayed only in the show running-config output if the MPLS MTU value is different from the interface MTU value. If the values match, only the interface MTU value is displayed.

If you attempt to set the MPLS MTU value higher than the interface MTU value, the software displays the following error message, which prompts you to set the interface MTU to a higher value before you set the MPLS MTU value:

```
% Please increase interface mtu to xxxx and then set mpls mtu
```

Note In Cisco IOS Release 15.1(2)S, the **mplsmtu** command was modified. This command was made available in L3VPN encapsulation configuration mode. The **maximum** keyword was replaced with the **max** keyword. The **override** keyword and the *bytes* argument were removed from the GRE tunnel interface. To set MPLS MTU to the maximum MTU on L3VPN profiles, use the **mplsmtu** command in L3VPN encapsulation configuration mode.

MPLS MTU Values for Ethernet Interfaces

If you have an interface with a default interface MTU value of 1580 or less (such as an Ethernet interface), the **mplsmtu** command provides an **override**keyword, which allows you to set the MPLS MTU to a value higher than the interface MTU value. The **override** keyword is not available for interface types that do not have a default interface MTU value of 1580 or less. For configuration details, see the Setting the MPLS MTU Value on an Ethernet Interface, page 51.

Setting the MPLS MTU value higher than the Ethernet interface MTU value can lead to dropped packets, data corruption, or high CPU rates. When you set the MPLS MTU value higher than the Ethernet interface MTU value, the software displays the following message:

```
%MFI-30-MPLS_MTU_SET: Setting the mpls mtu to xxxx on Ethernet x/x, which is higher than
the interface MTU xxxx. This could lead to packet forwarding problems including packet
drops.
Most drivers will be able to support baby giants and will gracefully drop packets that
are too large. Certain drivers will have packet forwarding problems including data
corruption.
Setting the mpls mtu higher than the interface mtu can lead to packet forwarding problems
and may be blocked in a future release.
```

Note The**override** keyword is supported in Cisco IOS Release 12.2(27)SBC, 12.2(33)SRA, 12.4(11)T, 12.2(33)SXH, and later releases, but may not be supported in a future release.

How to Configure MPLS MTU Values

The following sections explain how to configure MPLS MTU and interface MTU values:

- Setting the Interface MTU and MPLS MTU Values, page 50
- Setting the MPLS MTU Value on an Ethernet Interface, page 51
- Setting the MPLS MTU Value to the Maximum on L3VPN Profiles, page 52

Setting the Interface MTU and MPLS MTU Values

Use the following steps to set the interface MTU and the MPLS MTU.

Note In Cisco IOS Release 15.0(1)M1, you can configure the interface MTU on PA-1FE and PA-2FE port adapters. The range of values is from 1500 to 1530. Before this enhancement, the MTU of those interfaces was not configurable.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot / port*
4. **mtu** *bytes*
5. **mpls mtu** *bytes*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type slot / port* **Example:** `Router(config)# interface Serial 1/0`	Enters interface configuration mode to configure the interface.
Step 4	**mtu** *bytes* **Example:** `Router(config-if)# mtu 1520`	Sets the interface MTU size.

	Command or Action	Purpose
Step 5	**mpls mtu** *bytes* **Example:** `Router(config-if)# mpls mtu 1520`	Sets the MPLS MTU to match the interface MTU.
Step 6	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Setting the MPLS MTU Value on an Ethernet Interface

Use the following steps to set the MPLS MTU value on an Ethernet interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot / port*
4. **mpls mtu override** *bytes*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type slot / port* **Example:** `Router(config)# interface ethernet 1/0`	Enters interface configuration mode to configure the Ethernet interface.

	Command or Action	Purpose
Step 4	**mpls mtu override** *bytes* **Example:** `Router(config-if)# mpls mtu override 1510`	Sets the MPLS MTU to a value higher than the interface MTU value. **Caution** Setting the MPLS MTU to a value higher than the Ethernet interface MTU value can lead to dropped packets, data corruption, or high CPU rates.
Step 5	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Setting the MPLS MTU Value to the Maximum on L3VPN Profiles

Use the following steps to set the MPLS MTU value to the maximum on L3VPN profiles.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l3vpn encapsulation ip** *profile*
4. **mpls mtu max**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l3vpn encapsulation ip** *profile* **Example:** `Router(config)# l3vpn encapsulation ip profile1`	Configures an L3VPN encapsulation profile and enters the L3VPN encapsulation configuration mode.

	Command or Action	Purpose
Step 4	**mpls mtu max** **Example:** `Router(config-l3vpn-encap-ip)# mpls mtu max`	Sets the MPLS MTU value to the maximum MTU on the L3VPN profile.
Step 5	**end** **Example:** `Router(config-l3vpn-encap-ip)# end`	Exits L3VPN encapsulation configuration mode and returns to privileged EXEC mode.

Configuration Examples for Setting the MPLS MTU Values

Example Setting the Interface MTU and MPLS MTU

The following example shows how to set the interface and MPLS MTU values. The serial interface in the following configuration examples is at the default interface MTU value. The MPLS MTU value is not set. The interface settings are as follows:

```
interface Serial 4/0
 ip unnumbered Loopback0
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

The following example attempts to set the MPLS MTU value to 1520. This returns an error because MPLS MTU cannot be set to a value greater than the value of the interface MTU.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface serial 4/0
Router(config-if)# mpls mtu 1520
% Please increase interface mtu to 1520 and then set mpls mtu
```

The following example first sets the interface MTU to 1520 and then sets the MPLS MTU to 1520:

```
Router(config-if)# mtu 1520
Router(config-if)# mpls mtu 1520
```

The following example shows the new interface MTU value. The MPLS MTU value is not displayed because it is equal to the interface value.

```
Router#
```

```
show running-config interface serial 4/0
Building configuration...
interface Serial4/0
 mtu 1520
 ip unnumbered Loopback0
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

The following example sets the MPLS MTU value to 1510:

```
Router(config-if)# mpls mtu 1510
```

The following example shows the new interface MTU value. The MPLS MTU value is displayed because it is different than the interface MTU value.

```
Router#
show running-config interface serial 4/0
Building configuration...
interface Serial4/0
 mtu 1520
 ip unnumbered Loopback0
 mpls mtu 1510
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

Example Setting the MPLS MTU Value on an Ethernet Interface

Caution Setting the MPLS MTU to a value higher than the Ethernet interface MTU value can lead to dropped packets, data corruption, or high CPU rates.

The following example shows how to set the MPLS MTU values on an Ethernet interface. The Ethernet interface in the following configuration examples is at the default interface MTU value. The MPLS MTU value is not set. The interface settings are as follows:

```
interface Ethernet 2/0
 ip unnumbered Loopback0
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

The following example uses the **override** keyword to set the MPLS MTU to 1520, which is higher than the Ethernet interface's MTU value:

```
Router(config-if)# mpls mtu override 1520
%MFI-30-MPLS_MTU_SET: Setting the mpls mtu to 1520 on Ethernet2/0, which is higher than
the interface MTU 1500. This could lead to packet forwarding problems including packet
drops.
```

The following example shows the new MPLS MTU value:

```
Router#
show running-config interface ethernet 2/0
Building configuration...
interface Ethernet 2/0
 mtu 1500
```

```
 ip unnumbered Loopback0
 mpls mtu 1520
 mpls traffic-eng tunnels
 mpls ip
 serial restart-delay 0
 ip rsvp bandwidth 2000 2000
end
```

Example Setting the MPLS MTU Value to the Maximum MTU on L3VPN profiles

The following example shows how to set the MPLS MTU value to the maximum MTU on L3VPN profiles:

```
Router# configure terminal
Router(config)# l3vpn encapsulation ip profile1
Router(config-l3vpn-encap-ip)# mpls mtu max
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
MPLS commands	

MIBs

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

Technical Assistance

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

Feature Information for MPLS MTU Command Changes

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

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

Table 11 Feature Information for MPLS MTU Command Changes

Feature Name	Releases	Feature Information
MPLS MTU Command Changes	12.2(27)SBC 12.2(28)SB 12.2(33)SRA 12.2(33)SXH 12.4(11)T 15.0(1)M1 15.1(2)S	This document explains the changes to the **mplsmtu** command. You cannot set the MPLS MTU value larger than the interface MTU value, except for Ethernet interfaces. In 12.2(28)SB, support was added for the Cisco 10000 router. In 12.2(33)SRA, support was added for the Cisco 7600 series router. This feature was integrated into Cisco IOS Release 12.4(11)T. This feature was integrated into Cisco IOS Release 12.2(33)SXH. In 15.0(1)M1, you can configure the interface MTU on PA-1FE and PA-2FE port adapters. In 15.1(2)S, the **mplsmtu** command was made available in L3VPN encapsulation configuration mode. The **maximum** keyword was replaced with the **max** keyword. The **override** keyword and the *bytes* argument were removed from the GRE tunnel interface.

NetFlow MPLS Label Export

The NetFlow MPLS Label Export feature allows a label switch router (LSR) to collect and export Multiprotocol Label Switching (MPLS) labels allocated by the LSR when an adjacent router pushes that label on the top of the label stack of a transit packet. At the same time, the LSR collects the prefix associated with the MPLS label and the application that allocated the label. The router collects the information in a table called the MPLS Prefix/Application/Label (PAL) table and exports this data to a NetFlow collector as the label is allocated or, if so configured, periodically exports the full MPLS PAL table.

You can use this information to create a provider edge (PE)-to-PE matrix, which is useful for network traffic planning and billing. To realize this benefit, you must export the MPLS label information to a NetFlow collector for analysis. This feature also provides information that a NetFlow collector can use to create a Virtual Private Network (VPN) routing and forwarding instance (VRF)-to-PE and PE-to-VRF matrix.

Finding Feature Information

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

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

Prerequisites for NetFlow MPLS Label Export

The NetFlow MPLS Label Export feature requires the following:

- NetFlow configured on the LSR
- MPLS enabled on the LSR

If you are exporting data to a Cisco NetFlow collector, the following requirements apply:

- NetFlow Version 9 export format configured on the LSR
- NetFlow collector and analyzer that can use MPLS PAL records exported in NetFlow Version 9 format

Restrictions for NetFlow MPLS Label Export

The following restrictions apply to the NetFlow MPLS Label Export feature for Cisco IOS 12.2S releases and Cisco IOS Release 12.5(1):

- The MPLS PAL table does not support the export of information for the following:
  - IP Version 6 (IPv6) labels
  - IP Multicast labels
  - Quality of service (QoS) labels
  - Traffic engineering (TE) tunnel headend labels
- The ability to create a VRF-to-VRF traffic matrix is not supported.
- If one application deallocates a label and a second application soon reallocates the same label, the NetFlow collector might not be able to determine how many packets flowed while the label was owned by each application.
- In MPLS PAL table records, for labels allocated by VPNs, Border Gateway Protocol (BGP) IPv4, or BGP VPN Version 4 (VPNv4), the stored prefix can be either 0.0.0.0 or a route distinguisher (RD)-specific address:
  - If you do not configure the **mplsexportvpnv4prefixes** command, VPN prefixes are not tracked in the MPLS PAL table. These prefixes are displayed by the **showmplsflowmappings** command as 0.0.0.0.
  - If you configure the **mplsexportvpnv4prefixes**command, VPN prefixes are tracked and RD-specific addresses are displayed by the **showmplsflowmappings** command.

Information About NetFlow MPLS Label Export

The following sections contain useful information for understanding how to configure and use the NetFlow MPLS Label Export feature:

- MPLS Label Information Gathering and Exporting, page 60
- Labels Allocated by VPNs BGP IPv4 or BGP VPNv4 in the MPLS PAL Table, page 61
- MPLS PAL Table Record Export, page 62
- MPLS PAL and NetFlow Statistics Correlation on a NetFlow Collector, page 64
- MPLS Label Mapping on a Line Card, page 64

MPLS Label Information Gathering and Exporting

In a Cisco IOS 12.0S, 12.3T, or 12.4T release that supports the MPLS-Aware NetFlow feature, the mapping of the MPLS label to a prefix and an MPLS application is achieved through the use of the Label

Forwarding Information Base (LFIB). You can display this information with the**showipcacheverboseflow**command. These releases do not support the NetFlow MPLS Label Export feature.

In a Cisco IOS 12.2(28)SB release or later release that supports the NetFlow MPLS Label Export feature, the mapping of the MPLS label to a destination prefix or Forwarding Equivalence Class (FEC) and to the MPLS application currently using the label is achieved through the use of an MPLS PAL table. Each supported MPLS application on the router where the NetFlow MPLS Label Export feature is configured registers its label values, prefixes, and owning applications as the labels are allocated. This label-tracking functionality operates on the Route Processor (RP) software.

The MPLS label information (label to prefix and application) mapping is exported to a NetFlow collector at the time when the label is allocated. You can configure periodic export of the full MPLS PAL table to a collector for further processing and analysis through the use of the**mplsexportinterval**command.

An *interval* argument to the**mplsexportinterval** command controls the time in minutes between full MPLS PAL table exports to the NetFlow collector. You can configure an interval in the range of 0 to 10080 (1 week) minutes:

- If you want to export MPLS PAL table information only when the label is allocated, then configure this command with a 0 time interval with the **mplsexportinterval0**command.
- If you want to trigger an immediate export of the full MPLS PAL table, reconfigure the command with an *interval* argument that is different from the interval that is configured. For example, if you have configured the **mplsexportinterval1440**command, reconfigure the command with any nonzero number except 1440.
- If you have a complex network that generates a large amount of traffic, configure a large interval between MPLS PAL table exports. You might want to configure an interval from 6 to 12 hours (360 and 720 minutes).

The *interval* argument that you specify is the least amount of time that passes before another export of the MPLS PAL table occurs. The system could delay the MPLS PAL table export for 10 minutes if the PAL export queue already contains a large number of entries. This could happen if the export occurred at a time when thousands of routes just came up, or if NetFlow did not have the time to clear the export queue from either a previous export of the full table or a previous time when thousands of routes came up in a brief period of time.

After you have entered the **mplsexportinterval** command, you can use the **showmplsflowmappings**command to display MPLS PAL table entries. To display information about the number of MPLS PAL records exported to the collector, use the **showipflowexportverbose**command.

Labels Allocated by VPNs BGP IPv4 or BGP VPNv4 in the MPLS PAL Table

If you want to see VPN prefix information, that is, labels allocated by VPN, BGP IPv4, or BGP VPNv4, you need to configure the **mplsexportvpnv4prefixes** command. If you do not configure the **mplsexportvpnv4prefixes** command, MPLS PAL stores labels allocated by these application as prefix 0.0.0.0.

After you configure the **mplsexportvpnv4prefixes**command, the VPN prefix and the associated RD are stored in the MPLS PAL table. VPN addresses are made unique by adding an RD to the front of the address. The RD removes any ambiguity when the same VPN prefix is used for more than one VRF.

Note To export VPN prefixes and associated RDs from the MPLS PAL table, the first time you configure the **mplsexportvpnv4prefixes**command you need to save the configuration and reboot the router or clear all routes from the table.

To display the VPN prefix entries in the MPLS PAL table, use the **showmplsflowmappings**command.

With the **mplsexportvpnv4prefixes**command configured, a line of the output might look like this:

```
Router# show mpls flow mappings
Label    Owner    Route-Distinguisher Prefix           Allocated
.
.
.
27       BGP      100:1               10.34.0.0         00:57:48
```

The format of the Route-Distinguisher field in the output depends on how the RD was configured. The RD can be configured in the as-number:network number (ASN:nn) format, as shown in the example, or it can be configured in the IP address:network number format (IP-address:nn).

If you did not configure the **mplsexportvpnv4prefixes**command, a line of the output looks like this:

```
Router# show mpls flow mappings
.
.
.
Label    Owner    Route-Distinguisher Prefix           Allocated
21       BGP                          0.0.0.0          00:52:18
```

The Route-Distinguisher field is not populated and the Prefix is displayed as 0.0.0.0.

If the MPLS PAL table tracks a per-VRF aggregate label and you configured the **mplsexportvpnv4prefixes**command, the **showmplsflowmappings**command displays the RD associated with the per-VRF aggregate label, but the prefix for the per-VRF aggregate label is reported as 0.0.0.0. If the **mplsexportvpnv4prefixes**command is not configured, the per-VRF aggregate label is reported with no RD and prefix 0.0.0.0, and you cannot distinguish the per-VRF aggregate label from a normal BGP label.

MPLS PAL Table Record Export

In Cisco IOS Release 12.0S and later releases, the export of MPLS-Aware NetFlow cache records makes use of the NetFlow Version 9 export format data and template. The export of MPLS PAL table entries also uses the NetFlow Version 9 export format. MPLS PAL packets are exported as NetFlow options packets rather than NetFlow data packets. NetFlow options packets are defined in *Cisco Systems NetFlow Services Export Version 9,* Request for Comments (RFC) 3954.

The RP on the PE router learns and queues the MPLS PAL table records for export. The RP can combine large numbers of PAL table entries in a single Version 9 record and send the record to the NetFlow collector. The information exported by the RP contains instances of the following for each tracked label:

Label, allocating-application (Owner), Route-Distinguisher, Prefix, time stamp (Allocated)

Because the mapping may change as labels expire and are reused, each PAL record contains a time stamp indicating the system uptime at which the label was allocated.

NetFlow Export Template Format Used for MPLS PAL Entries

This is the NetFlow Version 9 export template format used for MPLS PAL entries:

MPLS label: 3 bytes

MPLS label application type: 1 byte

MPLS label IP prefix: 4 bytes

MPLS VPN prefix RD: 8 bytes

MPLS label allocation time: 4 bytes

MPLS Application Types Exported

The following MPLS application types are exported in the MPLS label application type field:

TE = 1

ATOM = 2

VPN = 3

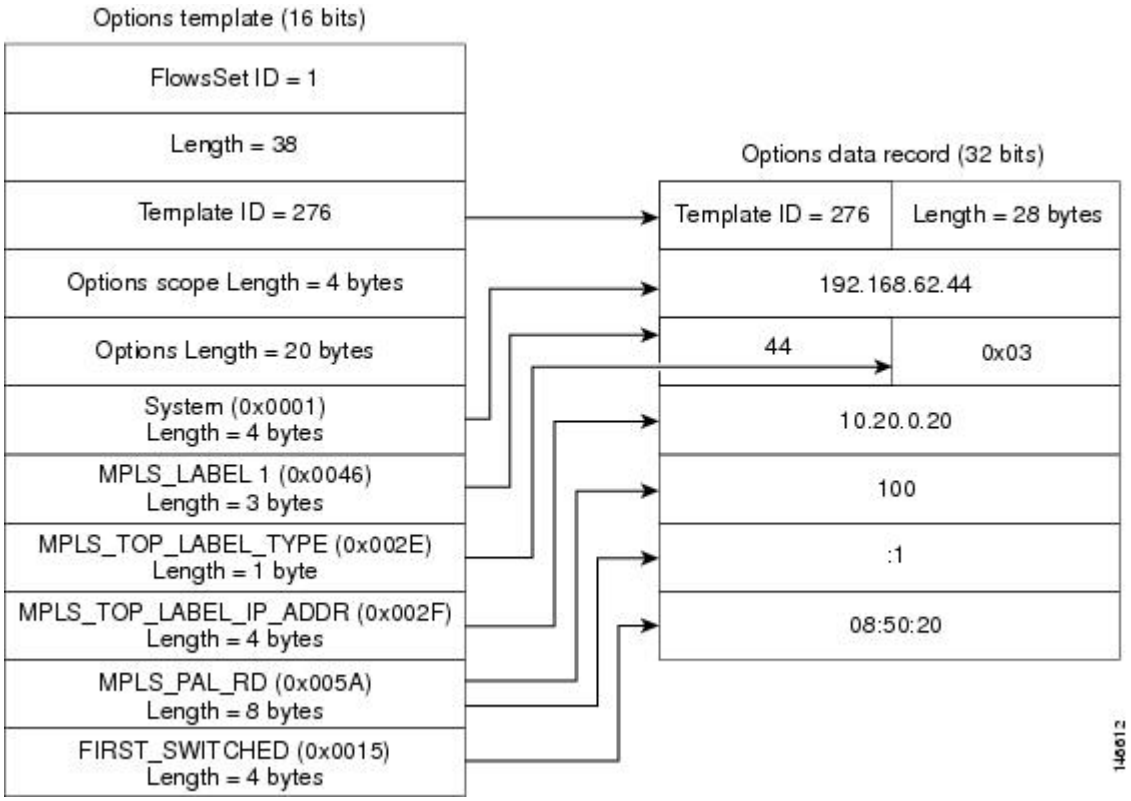
BGP = 4

LDP = 5

Options Template and Options Data Record for MPLS PAL Record Export

The figure below shows an example of the options template and options data record for MPLS PAL record export. This example shows that MPLS label 44 was allocated by a VPN 0x03 at 08:50:20 and is associated with the IP address 10.20.0.20 and with RD 100:1.

Figure 6 MPLS PAL Export Format Record



MPLS PAL and NetFlow Statistics Correlation on a NetFlow Collector

A NetFlow collector can gather the PAL NetFlow packets from a PE router and correlate the label mappings with the recent NetFlow records from adjacent provider core (P) routers.

For example, the MPLS PAL export packet contains MPLS label mappings over a period of time, as each label is allocated and reallocated on the PE router. The packet might contain the following information:

```
label 5, prefix 10.0.0.0,  type LDP, 12:00:00
label 4, prefix 10.10.0.0, type LDP, 13:00:00
label 5, prefix 10.9.0.0,  type VPN, 14:00:00
```

The NetFlow collector then receives a NetFlow packet from the adjacent P router indicating the following:

```
label 5, 123 packets, 9876 bytes, time 12:22:15.
```

The collector would match the time range known from the PAL packets with the line card (LC) packet time stamp. This would result in the correct mapping for label 5 at time 12:22:15, as follows:

```
label 5, application LDP, prefix 10.0.0.0.
```

The NetFlow collector needs to be able to handle relative differences in the time stamps caused by different reboot times of the P and PE routers.

To implement the offline label mapping checks in the NetFlow collector, the collector needs to maintain a history of label mappings obtained from the MPLS PAL NetFlow packets sent by the RP. If a label is deallocated and reallocated, the collector should track both the old and the new MPLS PAL information for the label.

Note On a rare occasion, the collector might not be able to accurately track how many packets flowed for a label that has been deallocated by one application and soon reallocated by another application.

MPLS Label Mapping on a Line Card

Label to prefix and application mapping is registered and exported from the router RP. This functionality does not occur on the line card. If you want to see the mapping for a particular label on a line card and the label of interest is tracked by the MPLS PAL table, then you can do the following:

- Enter the **showmplsforwarding** command on the line card.
- Enter the **showmplsflowmappings** on the RP.
- Compare the output of the two commands.

You might find the**|include**keyword to the commands useful in this case. For example, You could enter the**showmplsflowmappings|include777** command to see the information for any label with substring 777.

How to Configure NetFlow MPLS Label Export

Perform the following tasks to configure the NetFlow MPLS Label Export feature on an LSR. This feature provides the label, prefix, and application mapping through the MPLS PAL table that collects and exports the data to a NetFlow collector.

- Configuring NetFlow MPLS Label Export and MPLS PAL Table Export, page 65
- Displaying Information About the MPLS PAL Table, page 66

• Configuring the Export of MPLS VPN Version 4 Label Information from the MPLS PAL Table to a NetFlow Collector, page 68

Configuring NetFlow MPLS Label Export and MPLS PAL Table Export

Perform this task to configure the NetFlow MPLS Label Export feature and MPLS PAL table export to a NetFlow collector. You can use the information generated for network traffic planning and billing.

The following task must be completed before MPLS labels are allocated by the router for the MPLS PAL table to be exported to a NetFlow collector.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls export interval** *interval*
4. **end**
5. **copy running-config startup-config**
6. **exit**
7. Reboot the router.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**mpls export interval** *interval* **Example:** `Router(config)# mpls export interval 360` **Example:**	Configures a periodic time interval for the export of the entire MPLS PAL table to a NetFlow collector. • The *interval* argument specifies the time in minutes between full PAL table exports. The range of valid time intervals is 0 to 10,080 minutes. • We recommend that you select a time interval from 360 minutes (6 hours) to 1440 minutes (24 hours) depending on the size of your network and how often the NetFlow collector might be restarted. • If you enter an interval of 0, full PAL table exports are disabled. PAL information is exported only as labels are allocated. • If you need to restart your NetFlow collector and want to learn PAL information immediately, you can change the *interval* argument. When you change the time interval, the application exports the full PAL table. **Note** Allocated labels are tracked only after you enter the **mplsexportinterval** command. Any labels allocated before you enter this command are not tracked.
Step 4	**end** **Example:** `Router(config)#` **end**	Exits to privileged EXEC mode.
Step 5	**copy running-config startup-config** **Example:** `Router# copy running-config startup-config`	Copies the modified configuration into router NVRAM, permanently saving the settings. The next time the router is reloaded or rebooted the NetFlow MPLS Label Export feature is already part of the configuration.
Step 6	**exit** **Example:** `Router# exit`	Exits to user EXEC mode.
Step 7	Reboot the router.	(Optional) Saves the configuration and reboots the router to ensure that the information collected by this feature is complete.

Displaying Information About the MPLS PAL Table

Perform this task to display information about the MPLS PAL table. The information displayed includes the label, the application that allocated the label, an RD and destination prefix associated with the label, and the time the label was allocated by the application.

SUMMARY STEPS

1. **enable**
2. **show mpls flow mappings**
3. **show ip flow export verbose | include PAL**
4. **exit**

DETAILED STEPS

Step 1 **enable**

Use this command to enable privileged EXEC mode. Enter your password if prompted. For example:

Example:

```
Router> enable
Router#
```

Step 2 **show mpls flow mappings**

Use this command to display entries in the MPLS PAL table. For example:

Example:

```
Router# show mpls flow mappings
Label    Owner    Route-Distinguisher Prefix          Allocated
18       LDP                          10.0.0.5        00:52:10
21       BGP                          0.0.0.0         00:52:18
22       BGP                          0.0.0.0         00:52:18
25       BGP                          0.0.0.0         00:51:44
26       LDP                          10.32.0.0       00:52:10
27       TE-MIDPT                     10.30.0.2       00:52:06
28       LDP                          10.33.0.0       00:52:10
29       LDP                          10.0.0.1        00:52:10
30       LDP                          10.0.0.3        00:52:10
```

In this example, the **mplsexportvpnv4prefixes** command was not configured. Therefore, the MPLS PAL functionality did not export an RD for the BGP application, and the associated prefix is exported as 0.0.0.0.

The following shows sample output from the **showmplsflowmappings**command if you previously entered the **mplsexportvpnv4prefixes** command:

Example:

```
Router# show mpls flow mappings
Label    Owner    Route-Distinguisher Prefix          Allocated
16       LDP                          10.0.0.3        00:58:03
17       LDP                          10.33.0.0       00:58:03
19       TE-MIDPT                     10.30.0.2       00:58:06
20       LDP                          10.0.0.5        00:58:03
23       LDP                          10.0.0.1        00:58:03
24       LDP                          10.32.0.0       00:58:03
27       BGP      100:1               10.34.0.0       00:57:48
31       BGP      100:1               10.0.0.9        00:58:21
32       BGP      100:1               10.3.3.0        00:58:21
```

Step 3 **show ip flow export verbose | include PAL**

Use this command to display the number of MPLS PAL records that were exported to the NetFlow collector. For example:

Example:

```
Router# show ip flow verbose | include PAL
6 MPLS PAL records exported
```

When you specify the **verbose** keyword and MPLS PAL records have been exported using NetFlow Version 9 data format, the command output contains an additional line that precedes the "*x* records exported in *y* UDP datagrams" line.

Step 4 **exit**

Use this command to exit to user EXEC mode. For example:

Example:

```
Router# exit
Router>
```

Configuring the Export of MPLS VPN Version 4 Label Information from the MPLS PAL Table to a NetFlow Collector

Perform the following task to configure the export of VPNv4 label information from the MPLS PAL table to a NetFlow collector.

This allows you to track VPN prefix information for MPLS labels allocated by VPNs, BGP IPv4, and BGP VPNv4. You can use the data analyzed by the collector to assist in network traffic planning and billing.

A VRF must be configured on the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mpls export interval** *interval*
4. **mpls export vpnv4 prefixes**
5. **end**
6. **copy running-config startup-config**
7. **exit**
8. Reboot the router.
9. **enable**
10. **show mpls flow mappings**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**mpls export interval** *interval* **Example:** `Router(config)# mpls export interval 1440`	Configures the collection and export of MPLS PAL information to a NetFlow collector. • The *interval* argument specifies the time in minutes between full PAL table exports. The range of valid time intervals is 0 to 10,080 minutes. • We recommend that you select a time interval of 6 hours (360 minutes) to 24 hours (1440 minutes) depending on the size of your network. • If you enter an interval of 0, full PAL table exports are disabled. PAL information is exported only as labels are allocated. • If you need to restart your NetFlow collector and want to learn PAL information immediately, you can change the *interval* argument. When you change the time interval, the application exports the full PAL table.
Step 4	**mpls export vpnv4 prefixes** **Example:** `Router(config)# mpls export vpnv4 prefixes`	Configures the tracking and export of VPNv4 label information from the MPLS PAL table to a NetFlow collector.
Step 5	**end** **Example:** `Router(config)#` **end**	Exits to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** `Router# copy running-config startup-config`	Copies the modified configuration into router NVRAM, permanently saving the settings. The next time the router is rebooted the tracking and export of VPNv4 label information from the MPLS PAL table to a NetFlow collector is already part of the configuration.

	Command or Action	Purpose
Step 7	**exit** **Example:** `Router# exit`	Exits to user EXEC mode.
Step 8	Reboot the router.	(Optional) Saves the configuration and reboots the router to ensure that the information collected by this feature is complete.
Step 9	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 10	**show mpls flow mappings** **Example:** `Router# show mpls flow mappings`	Displays MPLS PAL table entries that include VPNv4 prefixes and VPN RDs.

Configuration Examples for NetFlow MPLS Label Export

Configuring NetFlow MPLS Prefix Application Label Table Export Examples

The following examples show how to configure NetFlow MPLS PAL table export on a PE router.

This example shows how to configure the export of the full MPLS PAL table every 480 minutes (8 hours):

```
configure terminal
!
mpls export interval 480
end
copy running-config startup-config
exit
```

This example shows how to configure MPLS PAL information export only as the labels are allocated:

```
configure terminal
!
mpls export interval 0
end
copy running-config startup-config
exit
```

In this example, the full MPLS PAL table is not exported repeatedly.

Configuring the Export of MPLS VPNv4 Label Information from the MPLS PAL Table Example

The following example shows how to configure the export of MPLS VPNv4 label information from the MPLS PAL table:

```
configure terminal
!
mpls export interval 720
mpls export vpnv4 prefixes
end
copy running-config startup-config
exit
```

The full MPLS PAL table with MPLS VPNv4 label information is configured to export to the NetFlow collector every 720 minutes (12 hours).

Additional References

The following sections provide references related to the NetFlow MPLS Label Export feature.

Related Documents

Related Topic	Document Title
Tasks for configuring MPLS-aware NetFlow	Configuring MPLS-aware NetFlow
Overview of the NetFlow application and advanced NetFlow features and services	Cisco IOS NetFlow Overview
Tasks for configuring NetFlow to capture and export network traffic data	Configuring NetFlow and NetFlow Data Export
Tasks for configuring MPLS egress NetFlow accounting	Configuring MPLS Egress NetFlow Accounting
Detailed information about the fields available in Version 9 export format and about export format architecture	Cisco IOS NetFlow Version 9 Flow-Record Format

Standards

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

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 3954	Cisco Systems NetFlow Services Export Version 9
RFC 2547	BGP/MPLS VPNs

Technical Assistance

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

Command Reference

The following commands are introduced or modified in the feature or features documented in this module. For information about these commands, see the *Cisco IOS Multiprotocol Label Switching Command Reference* at http://www.cisco.com/en/US/docs/ios/mpls/command/reference/mp_book.html. For information about all Cisco IOS commands, go to the Command Lookup Tool at http://tools.cisco.com/Support/CLILookup or to the *Cisco IOS Master Commands List* .

- **mpls export interval**
- **mpls export vpnv4 prefixes**
- **show ip flow export**
- **show mpls flow mappings**

Feature Information for NetFlow MPLS Label Export

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

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

Table 12 Feature Information for NetFlow MPLS Label Export

Feature Name	Releases	Feature Information
NetFlow MPLS Label Export	12.2(28)SB 12.2(33)SRA	The NetFlow MPLS Label Export feature provides the label switch router (LSR) with the capability of collecting and exporting the top label in the MPLS label stack along with its prefix or Forwarding Equivalence Class (FEC) and the application allocating the label to a NetFlow collector for supported MPLS applications. In 12.2(28)SB, this feature was introduced. In 12.2(33)SRA, this feature was integrated into a 12.2SRA release.

Glossary

BGP --Border Gateway Protocol. An interdomain routing protocol that replaces Exterior Gateway Protocol (EGP). A BGP system exchanges reachability information with other BGP systems. It is defined by RFC 1163.

export packet --A type of packet built by a device (for example, a router) with NetFlow services enabled. The packet is addressed to another device (for example, the NetFlow Collection Engine). The packet contains NetFlow statistics. The other device processes the packet (parses, aggregates, and stores information on IP flows).

FEC --Forward Equivalency Class. A set of packets that can be handled equivalently for the purpose of forwarding and thus is suitable for binding to a single label. The set of packets destined for an address prefix is one example of an FEC. A flow is another example

flow --A unidirectional stream of packets between a given source and destination--each of which is defined by a network-layer IP address and transport-layer source and destination port numbers. A unique flow is defined as the combination of the following key fields: source IP address, destination IP address, source port number, destination port number, Layer 3 protocol type, type of service (ToS), and input logical interface.

flowset --A collection of flow records that follow the packet header in an export packet. A flowset contains information that must be parsed and interpreted by the NetFlow Collection Engine. There are two different types of flowsets: template flowsets and data flowsets. An export packet contains one or more flowsets, and both template and data flowsets can be mixed in the same export packet.

IPv6 --IP Version 6. Replacement for IP Version 4 (IPv4). IPv6 includes support for flow ID in the packet header, which can be used to identify flows. Formerly called IPng (next generation).

label --A short, fixed-length identifier that tells switching nodes how the data (packets or cells) should be forwarded.

LDP --Label Distribution Protocol. A standard protocol that operates between Multiprotocol Label Switching (MPLS)-enabled routers to negotiate the labels (addresses) used to forward packets. The Cisco proprietary version of this protocol is the Tag Distribution Protocol (TDP).

LFIB --Label Forwarding Information Base. A data structure and way of managing forwarding in which destinations and incoming labels are associated with outgoing interfaces and labels.

LSR --label switch router. A router that forwards packets in a Multiprotocol Label Switching (MPLS) network by looking only at the fixed-length label.

MPLS --Multiprotocol Label Switching. A switching method in which IP traffic is forwarded through use of a label. This label instructs the routers and the switches in the network where to forward the packets. The forwarding of MPLS packets is based on preestablished IP routing information.

NetFlow --A Cisco IOS application that provides statistics on packets flowing through the router. It is emerging as a primary network accounting and security technology.

NetFlow Collection Engine (formerly NetFlow FlowCollector)--A Cisco application that is used with NetFlow on Cisco routers and Catalyst series switches. The NetFlow Collection Engine collects packets from the router that is running NetFlow and decodes, aggregates, and stores them. You can generate reports on various aggregations that can be set up on the NetFlow Collection Engine.

NetFlow v9 --NetFlow export format Version 9. A flexible and extensible means of carrying NetFlow records from a network node to a collector. NetFlow Version 9 has definable record types and is self-describing for easier NetFlow Collection Engine configuration.

network byte order --Internet-standard ordering of the bytes corresponding to numeric values.

options data record --Special type of data record that is used in the NetFlow process. It is based on an options template and has a reserved template ID that provides information about the NetFlow process itself.

options template --A type of template record that the router uses to communicate the format of NetFlow-related data to the NetFlow collector.

P router --provider core or backbone router. A router that is part of a service provider's core or backbone network and is connected to the provider edge (PE) routers.

packet header --First part of an export packet. It provides basic information about the packet (such as the NetFlow version, number of records contained in the packet, and sequence numbering) so that lost packets can be detected.

PAL table --Prefix/Application/Label table. A data structure that collects and exports the prefix, application, and time stamp for a specific label.

PE router --provider edge router. A router that is part of a service provider's network connected to a customer edge (CE) router. All Virtual Private Network (VPN) processing occurs in the PE router.

RD --route distinguisher. An 8-byte value that is concatenated with an IPv4 prefix to create a unique VPN IPv4 prefix.

There are two formats for configuring the route distinguisher argument. It can be configured in the as-number:network number (ASN:nn) format or it can be configured in the IP address:network number format (IP-address:nn).

RP --Route Processor. A processor module in the Cisco 7000 series routers that contains the CPU, system software, and most of the memory components that are used in the router. Sometimes called a Supervisory Processor.

TE --traffic engineering. Techniques and processes that cause routed traffic to travel through the network on a path other than the one that would have been chosen if standard routing methods were used.

TE tunnel --traffic engineering tunnel. A label-switched tunnel that is used for traffic engineering. Such a tunnel is set up through means other than normal Layer 3 routing; it is used to direct traffic over a path.

template flowset --A collection of template records that are grouped in an export packet.

template ID --A unique number that distinguishes a template record produced by an export device from other template records produced by the same export device. A NetFlow Collection Engine application can receive export packets from several devices. You should be aware that uniqueness is not guaranteed across export devices. Thus, you should configure the NetFlow Collection Engine to cache the address of the export device that produced the template ID in order to enforce uniqueness.

VPN --Virtual Private Network. A secure IP-based network that shares resources on one or more physical networks. A VPN contains geographically dispersed sites that can communicate securely over a shared backbone.

VPNv4 prefix --IPv4 prefix preceded by an 8-byte route distinguisher. The VPN addresses are made unique by adding a route distinguisher to the front of the address.