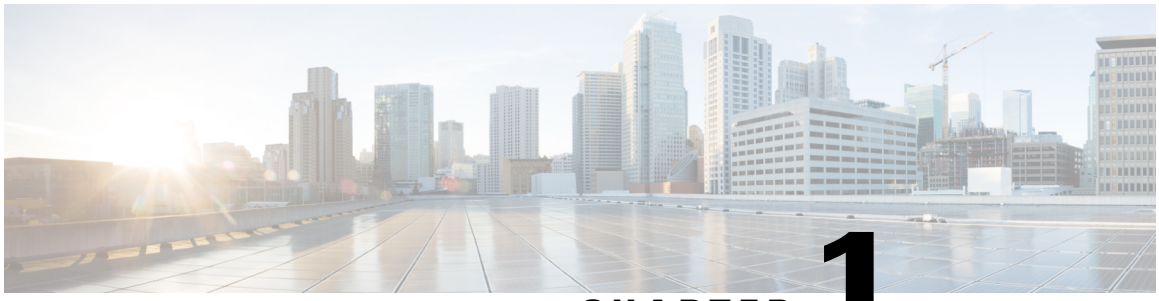
MPLS: Basic Configuration Guide, Cisco IOS Release 15SY

First Published: November 26, 2012

Last Modified: November 26, 2012

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



CHAPTER 1

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.

- MPLS Virtual Private Networks, page 9
- MPLS Quality of Service, page 9

Finding Feature Information

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

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to 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)

Old Tag Switching Terminology	New MPLS Terminology
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.

Command	Corresponding Tag Switching Command	Description
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.
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.

Command	Corresponding Tag Switching Command	Description
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.
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

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* .

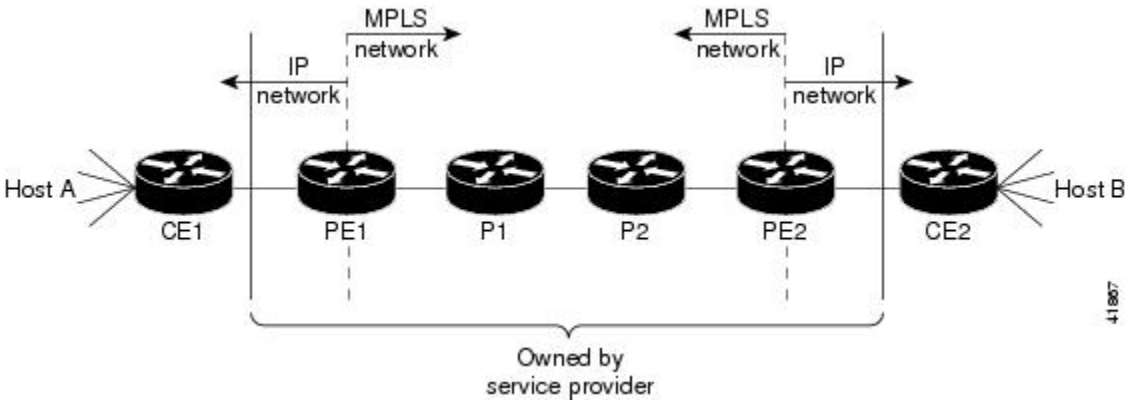
For complete command syntax information for CAR, WRED, and WFQ, refer to the *Cisco IOS Quality of Service Solutions Command Reference.*

Specifying the QoS in the IP Precedence Field

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.

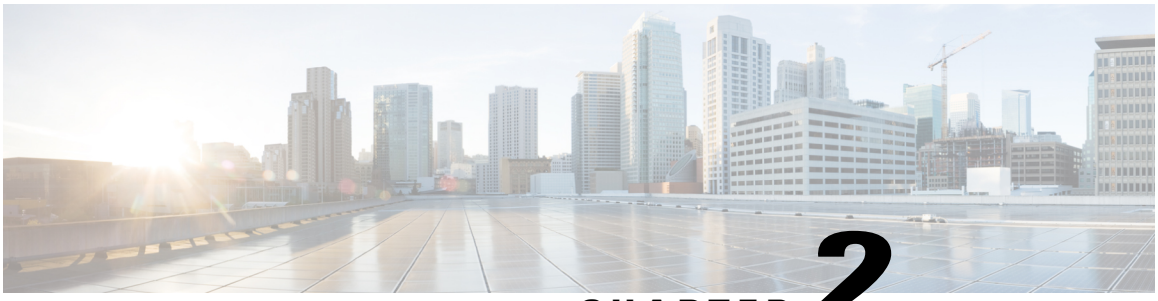
In the figure above, the following behavior occurs:

- 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.



CHAPTER 2

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 caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to 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, on page 17.

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

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.

	Command or Action	Purpose
Step 4	**mtu** *bytes* **Example:** `Router(config-if)# mtu 1520`	Sets the interface MTU size.
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.

	Command or Action	Purpose
Step 3	**interface** *type slot / port* **Example:** `Router(config)# interface ethernet 1/0`	Enters interface configuration mode to configure the Ethernet interface.
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.

	Command or Action	Purpose
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.
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

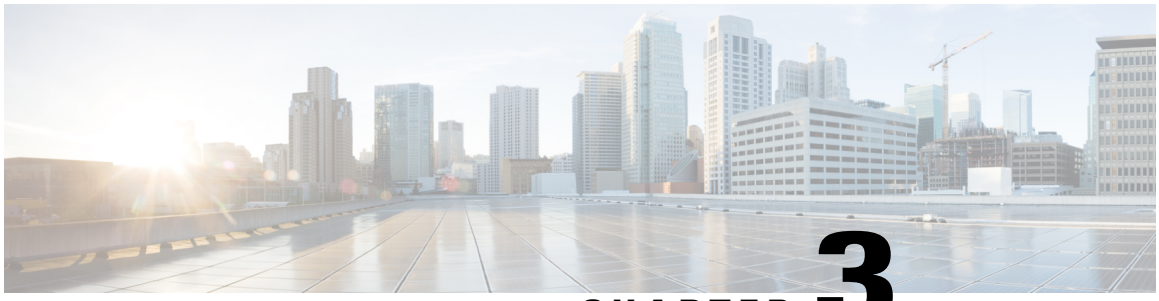
Feature Information for MPLS MTU Command Changes

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 5: 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.



CHAPTER 3

IP-Aware MPLS NetFlow

The IP-Aware MPLS NetFlow feature is an extension of the NetFlow accounting feature that uses copy-based sampling to copy sampled packets to the software where they can be further processed. This sampling provides highly granular traffic statistics for Cisco devices. NetFlow is a Cisco application that provides statistics about packets flowing through the device.

Finding Feature Information

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

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

Restrictions for Configuring IP-Aware MPLS NetFlow

- Copy-based sampling is allowed only in the ingress direction.
- The Lawful Intercept (LI) feature is of higher priority than the Capture2 (CAP2) feature. If LI is enabled, the copy-based sampling configuration is not removed but the feature is disabled.

- Inner Internet Protocol (IP) header meta details such as Ingress Interface, Egress Enterface, Border Gateway Protocol (BGP) next hop, and Interior Gateway Protocol (IGP) next hop are not allowed as collect fields with copy-based sampling.
- IP copy-based sampling supports both IP and Multiprotocol Label Switching (MPLS) packets coming in on the interface. However, these packets cannot be distinguished.
- All flow key and nonkey fields are enabled on the platform CLI but, during the configuration, only hardware supported fields are allowed on the noncopy-based sampling and nonsampling cases. For copy-based sampler all the fields are allowed, except the meta fields.
- The number of different profiles that can be used for copy-based sampling is limited to eight.
- In some cases the ingress and egress interface type cannot be derived in Cisco software.

Information About IP-Aware MPLS NetFlow

Benefits of the CAP2 Rate Limiter

During copy-based sampling the sampled packets are copied to the Route Processor. If this rate of sampling is high, the act of processing all of these packets in software may cause a negative impact on the CPU performance.

The CAP2 rate limiter limits the number of packets copied to the Route Processor, decreasing any chances of performance impact. The rate limiter configuration of access control list (ACL) logging (OAL) is also used for copy-based sampling rate limiting.

How to Configure IP-Aware MPLS NetFlow

Creating a Flow Record and Flow Exporter

Before You Begin

To enable copy-based sampling you must first create a flow record and flow exporter that can then be added to a flow monitor. The flow record is used for traffic analysis, and the exporter to export the data that is collected by flexible NetFlow.

Note Meta fields, such as number and BGP next hop, are not allowed with copy-based sampling.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **match ipv4 source** *address*
5. **end**
6. **flow exporter** *exporter-name*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow record** *record-name* **Example:** `Device(config)# flow record FLOW-RECORD-1`	Enters flexible NetFlow flow record configuration mode and creates a flow record.
Step 4	**match ipv4 source** *address* **Example:** `Device(config-flow-record)# match ipv4 source address`	Configures the IPv4 source address as a key field for the flow record.
Step 5	**end** **Example:** `Device(config-flow-record)# end`	Exits flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 6	**flow exporter** *exporter-name* **Example:** `Device(config)# flow exporter FLOW-EXPORTER-1`	Enters flexible NetFlow flow exporter configuration mode and creates a flow exporter.

	Command or Action	Purpose
Step 7	**end** **Example:** `Device(config-flow-exporter)# end`	Exits flexible NetFlow flow exporter configuration mode and returns to privileged EXEC mode.

Creating a Monitor and Adding a Flow Record and Flow Exporter

Before You Begin

To enable flow sampling, you configure the record that you want to use for traffic analysis, and the exporter to export the data that is collected by flexible NetFlow to a remote system for further analysis and storage, and assign them to a flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **record** *record-name*
5. **exporter** *exporter-name*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor FLOW-MONITOR-1`	Enters flexible NetFlow flow monitor configuration mode and creates a flow monitor.

	Command or Action	Purpose
Step 4	**record** *record-name* **Example:** `Device(config-flow-monitor)# record FLOW-RECORD-1`	Adds the record FLOW-RECORD-1 to the monitor.
Step 5	**exporter** *exporter-name* **Example:** `Device(config-flow-monitor)# exporter FLOW-EXPORTER-1`	Adds the exporter FLOW-EXPORTER-1 to the monitor.
Step 6	**end** **Example:** `Device(config-flow-monitor)# end`	Exits flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.

Configuring a Flow Sampler with a Copy Type

Before You Begin

Flow samplers are used to reduce the load placed by flexible NetFlow on the networking device to monitor traffic by limiting the number of packets that are analyzed. By applying the **copy type** command to the flow sampler, you enable the copying of sampled packets to the software or Route Processor. Features that are not available in hardware can then be applied on those packets.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **sampler** *sampler-name*
4. **type copy**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Device> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**sampler** *sampler-name* **Example:** `Device(config)# sampler SAMPLER-1`	Enters flexible NetFlow sampler configuration mode and creates a flow sampler with the name SAMPLER-1.
Step 4	**type copy** **Example:** `Device(config-sampler)# type copy`	Configures the sampler with the copy type to enable the sampled packets to be copied to Cisco software for accounting.
Step 5	**end** **Example:** `Device(config-sampler)# end`	Exits flexible NetFlow sampler configuration mode and returns to privileged EXEC mode.

Configuration Examples for IP-Aware MPLS NetFlow

Example: Creating a Flow Record and Flow Exporter

The following example shows how to create a flow record and flow exporter for copy-based sampling. Meta fields, such as number and BGP next hop, are not allowed with copy-based sampling.

```
Device(config)# flow record FLOW-RECORD-1
Device(config-record)# exit

Device(config)# flow exporter FLOW-EXPORTER-1
Device(config-exporter)#
```

Example: Configuring a Flow Monitor and Adding a Flow Record and Flow Exporter

The following configuration example, in flow monitor configuration mode, shows how to configure a flow monitor and add to it a flow record and flow exporter which enables flow sampling.

```
Device(config)# flow monitor FLOW-MONITOR-1
Device(config-flow-monitor)# record FLOW-RECORD-1
Device(config-flow-monitor)# exporter FLOW-EXPORTER-1
```

Example: Configuring a Sampler with a Copy Type

The following example shows how to configure a sampler for copy-based sampling. When the type **copy** is not specified the command is in noncopy-based sampling mode and the sampled packets are accounted for in hardware.

```
Device(config)# sampler SAMPLER-1
Device(config-sampler)# type copy
Device(config-sampler)# mode rand 1 out 10
Device(config)# end
```

Example: Applying the Monitor and Sampler to an Interface

The following example shows how to apply the monitor and sampler commands to an interface. Copy-based sampling in allowed only on the ingress direction. By applying a flow monitor and a sampler to the interface, you ensure the rate of analysis of the sampled packets is at the rate specified by the sampler. The sampled packets are then compared with the flow record associated with the flow monitor. If the analyzed packets meet the criteria specified by the flow record, they are added to the flow monitor cache.

```
Device(config)# interface g1/1
Device(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLER-1 input
```

Additional References for IP-Aware MPLS NetFlow

Related Documents

Related Topic	Document Title
Overview of Cisco IOS NetFlow	*Cisco IOS NetFlow Overview*
Cisco IOS commands	Master Commands List, All Releases

Technical Assistance

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

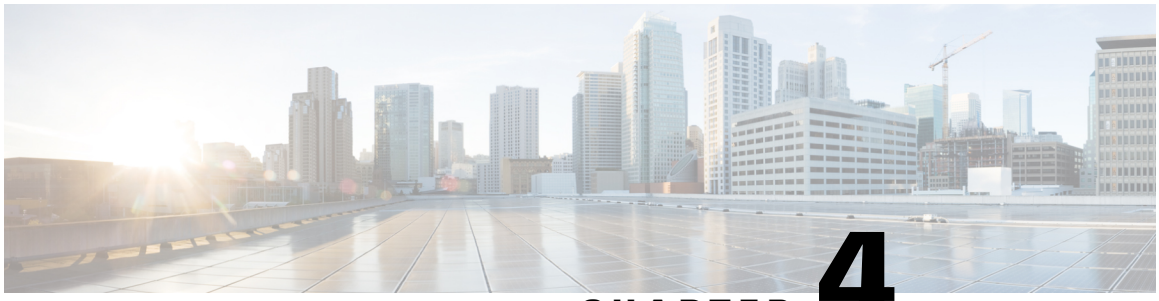
Feature Information for IP-Aware MPLS NetFlow

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 6: Feature Information for IP Aware MPLS Netflow

Feature Name	Releases	Feature Information
IP-Aware MPLS NetFlow	15.1(1)SY	The IP-Aware MPLS NetFlow feature is an extension of the NetFlow accounting feature that uses copy-based sampling to copy sampled packets to the software where they can be further processed. This sampling provides highly granular traffic statistics for Cisco devices. NetFlow is a Cisco application that provides statistics on packets flowing through the device. The following command was introduced: **type copy**.



CHAPTER 4

6PE Multipath

The 6PE multipath feature uses multiprotocol internal BGP (MP-iBGP) to distribute IPv6 routes over the MPLS IPv4 core network and to attach an MPLS label to each route

Finding Feature Information

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

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

Information About 6PE Multipath

6PE Multipath

Internal and external BGP multipath for IPv6 allows the IPv6 device to load balance between several paths (for example, the same neighboring autonomous system or subautonomous system, or the same metric) to reach its destination. The 6PE multipath feature uses MP-iBGP to distribute IPv6 routes over the MPLS IPv4 core network and to attach an MPLS label to each route.

When MP-iBGP multipath is enabled on the 6PE device, all labeled paths are installed in the forwarding table with MPLS information (label stack) when MPLS information is available. This functionality enables 6PE to perform load balancing.

How to Configure 6PE Multipath

Configuring IBGP Multipath Load Sharing

Perform this task to configure IBGP multipath load sharing and control the maximum number of parallel IBGP routes that can be installed in a routing table.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **maximum-paths ibgp** *number-of-paths*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** `Device(config)# router bgp 65000`	Enters router configuration mode for the specified routing process.
Step 4	**maximum-paths ibgp** *number-of-paths* **Example:** `Device(config-router)# maximum-paths ibgp 3`	Controls the maximum number of parallel IBGP routes that can be installed in a routing table.

Configuration Examples for 6PE Multipath

Example: Configuring 6PE Multipath

```
Device# show ipv6 cef internals
IPv6 CEF is enabled and running
Slow processing intvl = 1 seconds backoff level current/max 0/0
0 unresolved prefixes, 0 requiring adjacency update
IPv6 CEF default table
14 prefixes tableid 0
table version 17
root 6283F5D0
```

Additional References

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	*IPv6 Configuration Guide*
Cisco IOS commands	Master Commands List, All Releases
IPv6 commands	IPv6 Command Reference
Cisco IOS IPv6 features	IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	*IPv6 RFCs*

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

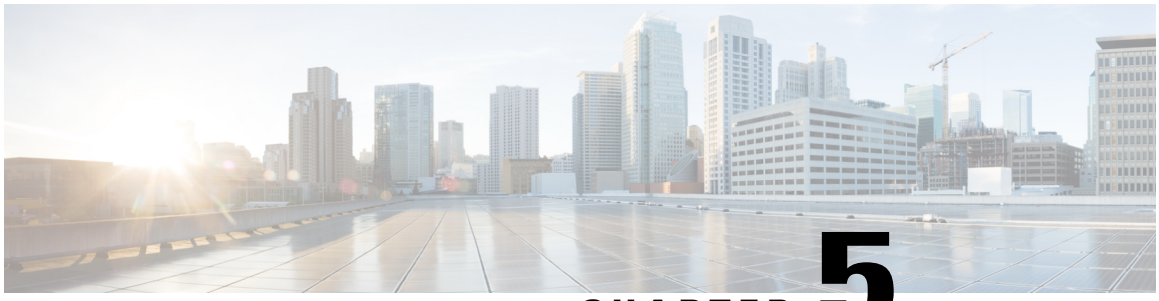
Feature Information for 6PE Multipath

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 7: Feature Information for 6PE Multipath

Feature Name	Releases	Feature Information
6PE Multipath	12.2(25)S 12.2(28)SB 12.2(33)SRA 12.4(6)T	The 6PE multipath feature uses MP-iBGP to distribute IPv6 routes over the MPLS IPv4 core network and to attach an MPLS label to each route. The following commands were introduced or modified: **maximum-paths ibgp**, **router bgp**, **show ipv6 cef internals**.



CHAPTER 5

IPv6 Switching: Provider Edge Device over MPLS

Multiprotocol Label Switching (MPLS) is deployed by many service providers in their IPv4 networks. Service providers want to introduce IPv6 services to their customers, but changes to their existing IPv4 infrastructure can be expensive and the cost benefit for a small amount of IPv6 traffic does not make economic sense. Several integration scenarios have been developed to leverage an existing IPv4 MPLS infrastructure and add IPv6 services without requiring any changes to the network backbone. This document describes how to implement IPv6 over MPLS.

Finding Feature Information

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

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

Prerequisites for IPv6 Switching: Provider Edge Device over MPLS

Before the IPv6 Provider Edge Device over MPLS (6PE) feature can be implemented, MPLS must be running over the core IPv4 network. If Cisco devices are used, Cisco Express Forwarding or distributed Cisco Express Forwarding must be enabled for both IPv4 and IPv6 protocols. This module assumes that you are familiar with MPLS.

Information About IPv6 Switching: Provider Edge Device over MPLS

Benefits of Deploying IPv6 over MPLS Backbones

IPv6 over MPLS backbones enables isolated IPv6 domains to communicate with each other over an MPLS IPv4 core network. This implementation requires only a few backbone infrastructure upgrades and no reconfiguration of core devices because forwarding is based on labels rather than the IP header itself, providing a very cost-effective strategy for the deployment of IPv6.

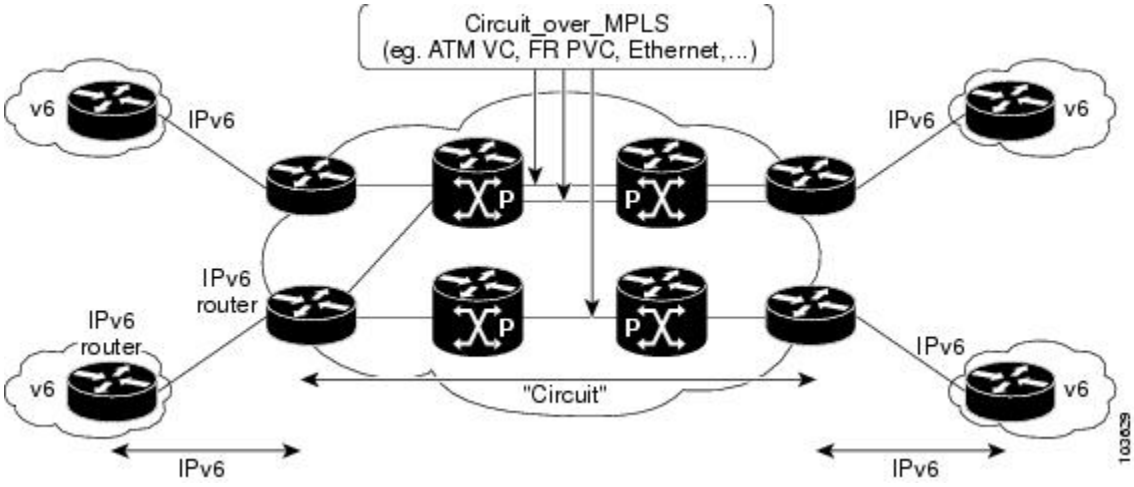
Additionally, the inherent Virtual Private Network (VPN) and MPLS traffic engineering (MPLS-TE) services available within an MPLS environment allow IPv6 networks to be combined into IPv4 VPNs or extranets over an infrastructure supporting IPv4 VPNs and MPLS-TE.

IPv6 over a Circuit Transport over MPLS

Using any circuit transport for deploying IPv6 over MPLS networks has no impact on the operation or infrastructure of MPLS, and requires no configuration changes to the core or provider edge devices. Communication between the remote IPv6 domains runs native IPv6 protocols over a dedicated link, where the underlying mechanisms are fully transparent to IPv6. The IPv6 traffic is tunneled using the Any Transport over MPLS (MPLS/AToM) or Ethernet over MPLS (EoMPLS) feature with the devices connected through an ATM OC-3 or Ethernet interface, respectively.

The figure below shows the configuration for IPv6 over any circuit transport over MPLS.

Figure 1: IPv6 over a Circuit Transport over MPLS

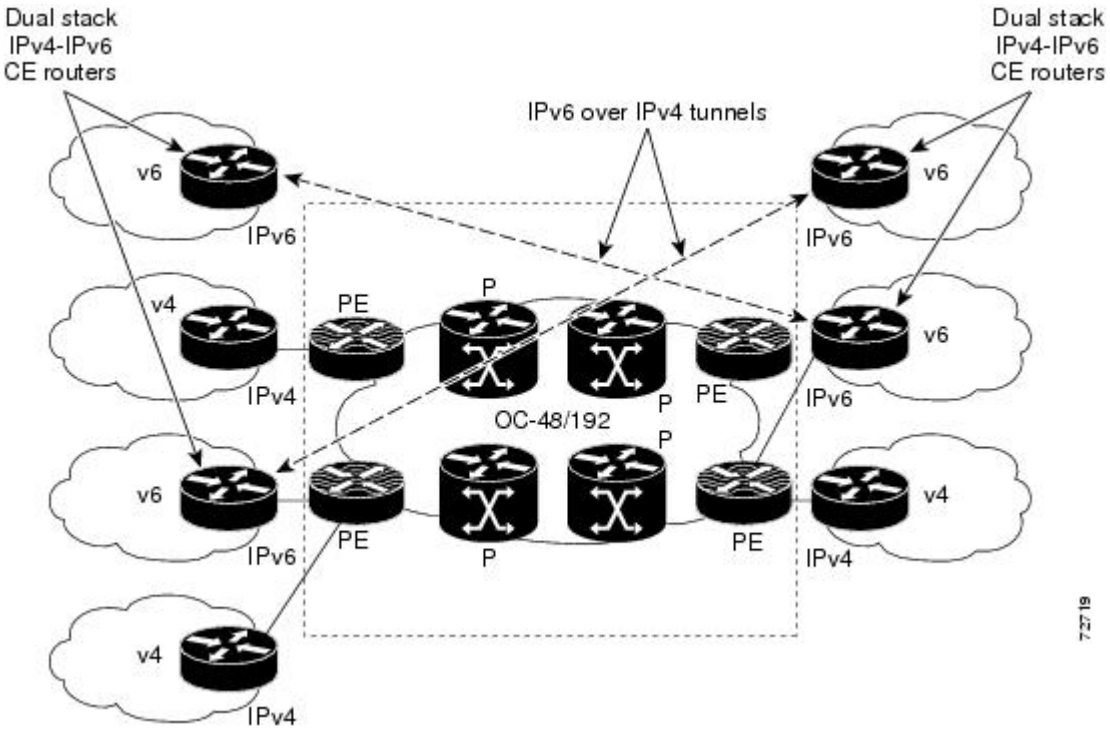


IPv6 Using Tunnels on the Customer Edge Devices

Using tunnels on the customer edge (CE) devices is the simplest way of deploying IPv6 over MPLS networks with no impact on the operation or infrastructure of MPLS, and no configuration changes to the core or provider edge devices. Communication between the remote IPv6 domains uses standard tunneling mechanisms and

requires the CE devices to be configured to run dual IPv4 and IPv6 protocol stacks. The figure below shows the configuration using tunnels on the CE devices.

Figure 2: IPv6 Using Tunnels on the CE Devices



Refer to Implementing Tunneling for IPv6 for configuration information on manually configured tunnels, automatic tunnels, and 6to4 tunnels.

Limitations on using tunnels involve the manual configuring of a mesh of tunnels on the CE devices, creating scaling issues for large networks.

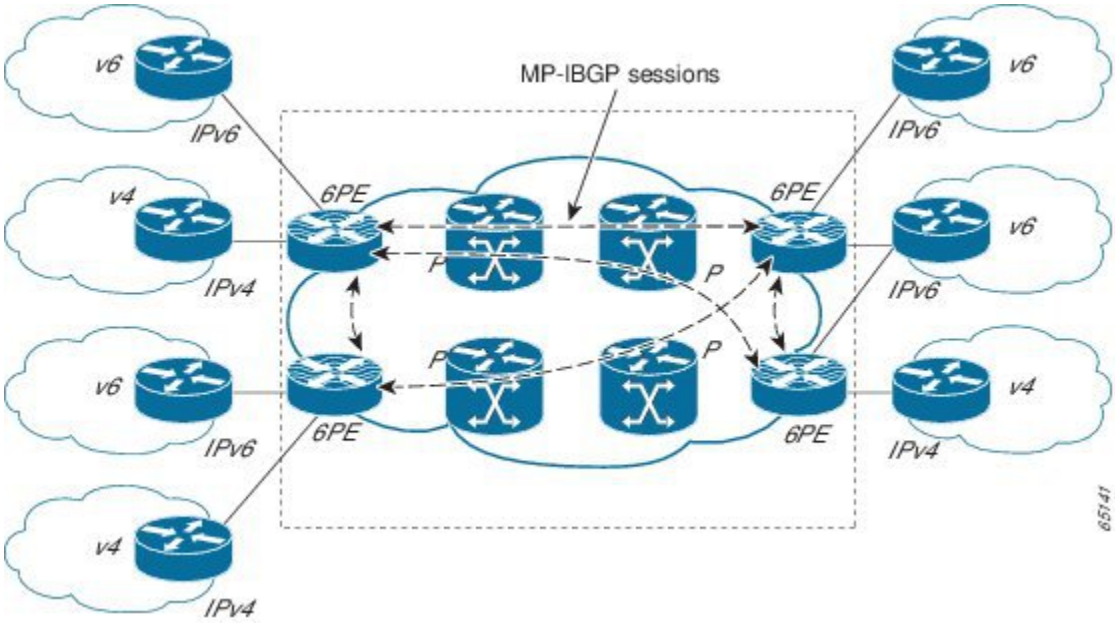
IPv6 on the Provider Edge Devices

The Cisco implementation of IPv6 provider edge device over MPLS is called 6PE, and it enables IPv6 sites to communicate with each other over an MPLS IPv4 core network using MPLS label switched paths (LSPs). This feature relies on multiprotocol Border Gateway Protocol (BGP) extensions in the IPv4 network configuration on the provider edge (PE) device to exchange IPv6 reachability information in addition to an MPLS label for each IPv6 address prefix to be advertised. Edge devices are configured to be dual stack running both IPv4 and IPv6, and use the IPv4 mapped IPv6 address for IPv6 prefix reachability exchange.

A hierarchy of labels is imposed on the 6PE ingress device to keep the IPv6 traffic transparent to all the core devices. The top label provides connectivity inside the IPv4 MPLS core network and the label is distributed by Label Distribution Protocol (LDP), Tag Distribution Protocol (TDP), or Resource Reservation Protocol (RSVP). TDP and LDP can both be used for label distribution, but RSVP is used only in the context of MPLS-TE label exchange. The bottom label, automatically assigned to the IPv6 prefix of the destination, is distributed by multiprotocol BGP and used at each 6PE egress device for IPv6 forwarding.

In the figure below the 6PE devices are configured as dual stack devices able to route both IPv4 and IPv6 traffic. Each 6PE device is configured to run LDP, TDP, or RSVP (if traffic engineering is configured) to bind the IPv4 labels. The 6PE devices use multiprotocol BGP to exchange reachability information with the other 6PE devices within the MPLS domain, and to distribute IPv6 labels between them. All 6PE and core devices--P devices in Figure 3--within the MPLS domain share a common IPv4 Interior Gateway Protocol (IGP) such as Open Shortest Path First (OSPF) or Integrated Intermediate System-to-Intermediate System (IS-IS).

Figure 3: 6PE Device Topology



The interfaces on the 6PE devices connecting to the CE device can be configured to forward IPv6 traffic, IPv4 traffic, or both types of traffic depending on the customer requirements. 6PE devices advertise IPv6 reachability information learned from their 6PE peers over the MPLS cloud. Service providers can delegate an IPv6 prefix from their registered IPv6 prefixes over the 6PE infrastructure; otherwise, there is no impact on the CE device.

The P devices in the core of the network are not aware that they are switching IPv6 packets. Core devices are configured to support MPLS and the same IPv4 IGP as the PE devices to establish internal reachability inside the MPLS cloud. Core devices also use LDP, TDP, or RSVP for binding IPv4 labels. Implementing the Cisco 6PE feature does not have any impact on the MPLS core devices.

Within the MPLS network, IPv6 traffic is forwarded using label switching, making the IPv6 traffic transparent to the core of the MPLS network. No IPv6 over IPv4 tunnels or Layer 2 encapsulation methods are required.

How to Deploy IPv6 Switching: Provider Edge Device over MPLS

Deploying IPv6 over a Circuit Transport over MPLS

To deploy IPv6 over a circuit transport over MPLS, the IPv6 devices must be configured for IPv6 connectivity. The MPLS device configuration requires AToM configuration or EoMPLS configuration.

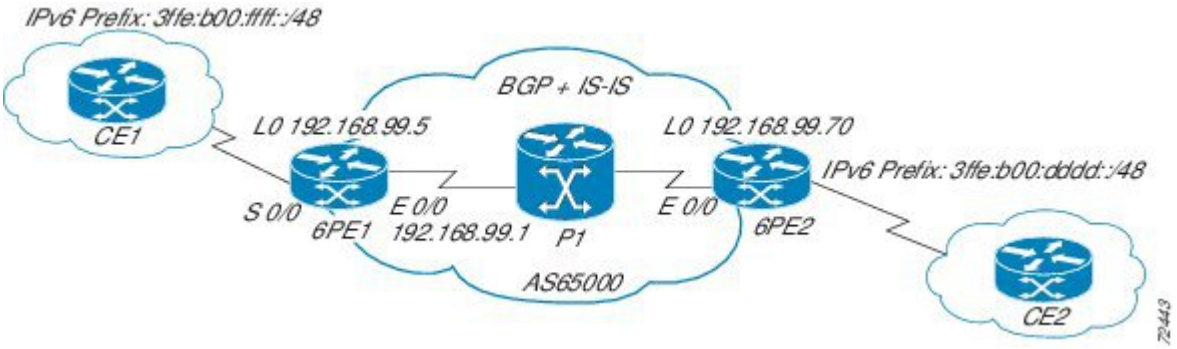
Deploying IPv6 on the Provider Edge Devices (6PE)

Specifying the Source Address Interface on a 6PE Device

Two configuration tasks using the network shown in the figure below are required at the 6PE1 device to enable the 6PE feature.

The customer edge device--CE1 in the figure below--is configured to forward its IPv6 traffic to the 6PE1 device. The P1 device in the core of the network is assumed to be running MPLS, a label distribution protocol, an IPv4 IGP, and Cisco Express Forwarding or distributed Cisco Express Forwarding, and does not require any new configuration to enable the 6PE feature.

Figure 4: 6PE Configuration Example



Before You Begin

- The 6PE devices--the 6PE1 and 6PE2 devices in the figure below--must be members of the core IPv4 network. The 6PE device interfaces attached to the core network must be running MPLS, the same label distribution protocol, and the same IPv4 IGP, as in the core network.
- The 6PE devices must also be configured to be dual stack to run both IPv4 and IPv6.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 unicast-routing**
4. **ipv6 cef**
5. **interface** *type number*
6. **ipv6 address** *ipv6-address / prefix-length | prefix-name sub-bits / prefix-length*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 unicast-routing** **Example:** `Device(config)# ipv6 unicast-routing`	Enables the forwarding of IPv6 unicast datagrams.
Step 4	**ipv6 cef** **Example:** `Device(config)# ipv6 cef`	Enables IPv6 Cisco Express Forwarding.
Step 5	**interface** *type number* **Example:** `Device(config)# interface Serial 0/0`	Specifies an interface type and number and enters interface configuration mode. • In the context of this feature, the interface to be configured is the interface communicating with the CE device.
Step 6	**ipv6 address** *ipv6-address / prefix-length | prefix-name sub-bits / prefix-length* **Example:** `Device(config-if)# ipv6 address 2001:DB8:FFFF::2/64`	Configures an IPv6 address based on an IPv6 general prefix and enable IPv6 processing on an interface.

Binding and Advertising the 6PE Label to Advertise Prefixes

Perform this task to enable the binding and advertising of labels when advertising IPv6 prefixes to a specified BGP neighbor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **no bgp default ipv4-unicast**
5. **neighbor** {*ip-address* | *ipv6-address* | *peer-group-name*} **remote-as** *as-number*
6. **neighbor** { *ip-address* | *ipv6-address* | *peer-group-name*} **update-source** *interface-type interface-number*
7. **address-family ipv6** [**unicast**]
8. **neighbor** {*ip-address* | *peer-group-name*| *ipv6-address*} **activate**
9. **neighbor** {*ip-address* | *ipv6-address*} **send-label**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** `Device(config)# router bgp 65000`	Enters router configuration mode for the specified routing process.
Step 4	**no bgp default ipv4-unicast** **Example:** `Device(config-router)# no bgp default ipv4-unicast`	Disables the IPv4 unicast address family for the BGP routing process specified in the previous step. **Note** Routing information for the IPv4 unicast address family is advertised by default for each BGP routing session configured with the **neighbor remote-as** command unless you configure the **no bgp default ipv4-unicast**command before configuring the **neighbor remote-as** command.
Step 5	**neighbor** {*ip-address* | *ipv6-address* | *peer-group-name*} **remote-as** *as-number* **Example:** `Device(config-router)# neighbor 192.168.99.70 remote-as 65000`	Adds the IP address of the neighbor in the specified autonomous system to the BGP neighbor table of the local device.

	Command or Action	Purpose
Step 6	**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name*} **update-source** *interface-type interface-number* **Example:** `Device(config-router)# neighbor 192.168.99.70 update-source Loopback 0`	Specifies the interface whose IPv4 address is to be used as the source address for the peering. • In the context of this task, the interface must have an IPv4 address with a 32-bit mask configured. Use of a loopback interface is recommended. This address is used to determine the IPv6 next hop by the peer 6PE.
Step 7	**address-family ipv6** [**unicast**] **Example:** `Device(config-router)# address-family ipv6`	Specifies the IPv6 address family and enters address family configuration mode. • The **unicast** keyword specifies the IPv6 unicast address family. By default, the device is placed in configuration mode for the IPv6 unicast address family if the **unicast** keyword is not specified with the **address-family ipv6** command.
Step 8	**neighbor** {*ip-address* | *peer-group-name*| *ipv6-address*} **activate** **Example:** `Device(config-router-af)# neighbor 192.168.99.70 activate`	Enables the neighbor to exchange prefixes for the IPv6 address family with the local device.
Step 9	**neighbor** {*ip-address* | *ipv6-address*} **send-label** **Example:** `Device(config-router-af)# neighbor 192.168.99.70 send-label`	Advertises the capability of the device to send MPLS labels with BGP routes. • In IPv6 address family configuration mode this command enables binding and advertisement of labels when advertising IPv6 prefixes in BGP.

Configuring IBGP Multipath Load Sharing

Perform this task to configure IBGP multipath load sharing and control the maximum number of parallel IBGP routes that can be installed in a routing table.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **maximum-paths ibgp** *number-of-paths*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** `Device(config)# router bgp 65000`	Enters router configuration mode for the specified routing process.
Step 4	**maximum-paths ibgp** *number-of-paths* **Example:** `Device(config-router)# maximum-paths ibgp 3`	Controls the maximum number of parallel IBGP routes that can be installed in a routing table.

Configuration Examples for IPv6 Switching: Provider Edge Device over MPLS

Example: Customer Edge Device

This example shows that the serial interface 0/0 of the customer edge device--CE1 in the figure above--is connected to the service provider and is assigned an IPv6 address. IPv6 is enabled and a default static route is installed using the IPv6 address of serial interface 0/0 of the 6PE1 device.

```
ip cef
!
ipv6 unicast-routing
!
interface Serial 0/0
 description to_6PE1_router
 no ip address
 ipv6 address 2001:DB8:FFFF::2/64
!
ipv6 route ::/0 Serial 0/0 FE80::210:XXXX:FEE1:1001
```

Example: Provider Edge Device

The 6PE device--Device 6PE1 in the figure above--is configured for both IPv4 and IPv6 traffic. Ethernet interface 0/0 is configured with an IPv4 address and is connected to a device in the core of the network--device P1 in the figure above. Integrated IS-IS and TDP configurations on this device are similar to the P1 device.

Device 6PE1 exchanges IPv6 routing information with another 6PE device--Device 6PE2 in the figure above--using internal BGP (iBGP) established over an IPv4 connection so that all the **neighbor** commands use the IPv4 address of the 6PE2 device. All the BGP peers are within autonomous system 65000, so synchronization with IGP is turned off for IPv4. In IPv6 address family configuration mode, synchronization is disabled by default.

IPv6 and Cisco Express Forwarding for IPv6 are enabled, the 6PE2 neighbor is activated, and label binding and advertisement is enabled for IPv6 prefixes using the **neighbor send-label** command. Connected and static IPV6 routes are redistributed using BGP. If IPv6 packets are generated in the local device, the IPv6 address for MPLS processing will be the address of loopback interface 0.

This example shows that the serial interface 0/0 connects to the customer and the IPv6 prefix delegated to the customer is 2001:DB8:ffff::/48, which is determined from the service provider IPv6 prefix. A static route is configured to route IPv6 packets between the 6PE route and the CE device.

```
ip cef
ipv6 cef
ipv6 unicast-routing
!
mpls ipv6 source-interface Loopback0
tag-switching tdp router-id Loopback0
!
interface Loopback0
 ip address 192.168.99.5 255.255.255.255
 ipv6 address 2001:DB8:1000:1::1/64
!
interface Ethernet0/0
 description to_P_router
 ip address 192.168.99.1 255.255.255.252
 ip router isis
 tag-switching ip
!
interface Serial0/0
 description to_CE_router
 no ip address
 ipv6 address 2001:DB8:FFFF::1/64
!
router isis
 passive-interface Loopback0
 net 49.0001.1921.6809.9005.00
!
router bgp 65000
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 neighbor 192.168.99.70 remote-as 65000
 neighbor 192.168.99.70 description to_6PE2
 neighbor 192.168.99.70 update-source Loopback0
 !
 address-family ipv6
 neighbor 192.168.99.70 activate
 neighbor 192.168.99.70 send-label
 network 2001:DB8:FFFF::/48
 exit-address-family
!
ipv6 route 2001:DB8:FFFF::/48 Ethernet0/0 2001:DB8:FFFF::2
```

Example: Core Device

This example shows that the device in the core of the network--Device P in the figure above--is running MPLS, IS-IS, and IPv4 only. The Ethernet interfaces are configured with IPv4 address and are connected to the 6PE devices. IS-IS is the IGP for this network and the P1 and 6PE devices are in the same IS-IS area 49.0001. TDP and tag switching are enabled on both the Ethernet interfaces. Cisco Express Forwarding is enabled in global configuration mode.

```
ip cef
!
tag-switching tdp router-id Loopback0
!
interface Loopback0
 ip address 192.168.99.200 255.255.255.255
!
interface Ethernet0/0
 description to_6PE1
 ip address 192.168.99.2 255.255.255.252
 ip router isis
 tag-switching ip
!
interface Ethernet0/1
 description to_6PE2
 ip address 192.168.99.66 255.255.255.252
 ip router isis
 tag-switching ip
router isis
 passive-interface Loopback0
 net 49.0001.1921.6809.9200.00
```

Example: Monitoring 6PE

This example shows output information about an IPv6 route using the **show bgp ipv6** command with an IPv6 prefix:

```
Device# show bgp ipv6 2001:DB8:DDDD::/48

BGP routing table entry for 2001:DB8:DDDD::/48, version 15
Paths: (1 available, best #1, table Global-IPv6-Table)
  Not advertised to any peer
  Local
    ::FFFF:192.168.99.70 (metric 20) from 192.168.99.70 (192.168.99.70)
      Origin IGP, localpref 100, valid, internal, best
```

This example shows output information about a BGP peer, including the IPv6 label capability, using the **show bgp ipv6 neighbors** command with an IP address:

```
Device# show bgp ipv6 neighbors 192.168.99.70

BGP neighbor is 192.168.99.70, remote AS 65000, internal link
  BGP version 4, remote router ID 192.168.99.70
  BGP state = Established, up for 00:05:17
  Last read 00:00:09, hold time is 0, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received
    Address family IPv6 Unicast: advertised and received
    ipv6 MPLS Label capability: advertised and received
  Received 54 messages, 0 notifications, 0 in queue
  Sent 55 messages, 1 notifications, 0 in queue
  Default minimum time between advertisement runs is 5 seconds

 For address family: IPv6 Unicast
  BGP table version 21880, neighbor version 21880
  Index 1, Offset 0, Mask 0x2
  Route refresh request: received 0, sent 0
```

```
  77 accepted prefixes consume 4928 bytes
  Prefix advertised 4303, suppressed 0, withdrawn 1328
  Number of NLRIs in the update sent: max 1, min 0
```

This example shows output information linking the MPLS label with prefixes using the **show mpls forwarding-table** command. If the 6PE feature is configured, the labels are aggregated because there are several prefixes for one local label, and the prefix column contains IPv6 instead of a target prefix.

```
Device# show mpls forwarding-table

Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         Pop Label  1.1.1.1/32       0             Et0/0      10.0.0.1
18         No Label   nh-id(1)         0             Et2/0      10.0.2.2
19         No Label   nh-id(2)         0             Et1/0      10.0.1.2
20         No Label   nh-id(3)         0             Et1/0      10.0.1.2
22         No Label   nh-id(5)         0             Et1/0      10.0.1.2
24         No Label   nh-id(5)         0             Et2/0      10.0.2.2
```

This example shows output information about the top of the stack label with label switching information using the **show bgp ipv6** command with the **labels** keyword:

```
Device# show bgp ipv6 labels

Network              Next Hop               In tag/Out tag
2001:DB8:DDDD::/64  ::FFFF:192.168.99.70   notag/20
```

This example shows output information about labels from the Cisco Express Forwarding table using the **show ipv6 cef** command with an IPv6 prefix:

```
Device# show ipv6 cef 2001:DB8:DDDD::/64

2001:DB8:DDDD::/64
      nexthop ::FFFF:192.168.99.70
      fast tag rewrite with Se0/0, point2point, tags imposed {19 20}
```

This example shows output information from the IPv6 routing table using the **show ipv6 route** command. The output shows the IPv6 MPLS virtual interface as the output interface of IPv6 routes forwarded across the MPLS cloud.

The 6PE2 device has advertised the IPv6 prefix of 2001:DB8:dddd::/48 configured for the CE2 device and the next-hop address is the IPv4-compatible IPv6 address ::ffff:192.168.99.70, where 192.168.99.70 is the IPv4 address of the 6PE2 device.

```
Device# show ipv6 route

IPv6 Routing Table - 10 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
B 2001:DB8:DDDD::/64 [200/0]
  via ::FFFF:192.168.99.70, IPv6-mpls
B 2001:DB8:DDDD::/64 [200/0]
  via ::FFFF:192.168.99.70, IPv6-mpls
L 2001:DB8:FFFF::1/128 [0/0]
  via ::, Ethernet0/0
C 2001:DB8:FFFF::/64 [0/0]
  via ::, Ethernet0/0
S 2001:DB8:FFFF::/48 [1/0]
  via 2001:DB8:B00:FFFF::2, Ethernet0/0
```

Additional References for IPv6 Switching: Provider Edge Router over MPLS

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	*IPv6 Configuration Guide*
Cisco IOS commands	Master Commands List, All Releases
IPv6 commands	IPv6 Command Reference
Cisco IOS IPv6 features	IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	*IPv6 RFCs*

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 IPv6 Switching: Provider Edge Device over MPLS

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 8: Feature Information for IPv6 Switching: Provider Edge Device over MPLS

Feature Name	Releases	Feature Information
IPv6 Switching: Provider Edge Router over MPLS	12.0(22)S 12.2(14)S 12.2(28)SB 12.2(33)SRA 12.2(15)T	The Cisco implementation of IPv6 provider edge device over MPLS enables IPv6 sites to communicate with each other over an MPLS IPv4 core network using MPLS LSPs. The following commands were introduced or modified: **address-family ipv6**, **ipv6 address**, **ipv6 cef**, **ipv6 unicast-routing**, **maximum-paths ibgp**, **neighbor activate**, **neighbor remote-as**, **neighbor send-label**, **neighbor update-source**, **no bgp default ipv4-unicast**, **router bgp**, **show bgp ipv6**, **show bgp ipv6 labels**, **show bgp ipv6 neighbors**, **show ipv6 cef**, **show ipv6 route**, **show mpls forwarding-table**.