CONFIGURATION GUIDE

Extreme SLX-OS MPLS Configuration Guide, 17r.2.01

Supporting the ExtremeRouting SLX 9850 and ExtremeSwitching 9540 Devices

9035316-01
15 June 2018

Contents

Preface

Document conventions

The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in Extreme technical documentation.

Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

NOTE
A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

CAUTION
A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER
A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used to highlight specific words or phrases.

Format	Description
bold text	Identifies command names.
	Identifies keywords and operands.
	Identifies the names of GUI elements.
	Identifies text to enter in the GUI.
italic text	Identifies emphasis.
	Identifies variables.
	Identifies document titles.
`Courier font`	Identifies CLI output.

Format	Description
	Identifies command syntax examples.

Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{ **x** | **y** | **z** }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x | **y**	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*[*member*...].
\	Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Extreme resources

Visit the Extreme website to locate related documentation for your product and additional Extreme resources.

White papers, data sheets, and the most recent versions of Extreme software and hardware manuals are available at www.extremenetworks.com. Product documentation for all supported releases is available to registered users at www.extremenetworks.com/support/documentation.

Document feedback

Quality is our first concern at Extreme, and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you.

You can provide feedback in two ways:

- Use our short online feedback form at http://www.extremenetworks.com/documentation-feedback-pdf/
- Email us at internalinfodev@extremenetworks.com

Provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.

Contacting Extreme Technical Support

As an Extreme customer, you can contact Extreme Technical Support using one of the following methods: 24x7 online or by telephone. OEM customers should contact their OEM/solution provider.

If you require assistance, contact Extreme Networks using one of the following methods:

- GTAC (Global Technical Assistance Center) for immediate support
  - Phone: 1-800-998-2408 (toll-free in U.S. and Canada) or +1 408-579-2826. For the support phone number in your country, visit: www.extremenetworks.com/support/contact.
  - Email: support@extremenetworks.com. To expedite your message, enter the product name or model number in the subject line.
- GTAC Knowledge - Get on-demand and tested resolutions from the GTAC Knowledgebase, or create a help case if you need more guidance.
- The Hub - A forum for Extreme customers to connect with one another, get questions answered, share ideas and feedback, and get problems solved. This community is monitored by Extreme Networks employees, but is not intended to replace specific guidance from GTAC.
- Support Portal - Manage cases, downloads, service contracts, product licensing, and training and certifications.

Before contacting Extreme Networks for technical support, have the following information ready:

- Your Extreme Networks service contract number and/or serial numbers for all involved Extreme Networks products
- A description of the failure
- A description of any action(s) already taken to resolve the problem
- A description of your network environment (such as layout, cable type, other relevant environmental information)
- Network load at the time of trouble (if known)
- The device history (for example, if you have returned the device before, or if this is a recurring problem)
- Any related RMA (Return Material Authorization) numbers

About This Document

Supported hardware and software

In those instances in which procedures or parts of procedures documented here apply to some devices but not to others, this guide identifies exactly which devices are supported and which are not.

Although many different software and hardware configurations are tested and supported by Extreme Networks for SLX-OS Release 17r.2.01, documenting all possible configurations and scenarios is beyond the scope of this document.

The following hardware platforms are supported by this release:

- ExtremeRouting SLX 9850-4 router
- ExtremeRouting SLX 9850-8 router
- ExtremeSwitching SLX 9540 switch

To obtain information about other releases, refer to the documentation specific to that release.

Interface module capabilities

The following table lists the supported capabilities for the following SLX 9850 interface modules:

- BR-SLX9850-10Gx72S-M
- BR-SLX9850-100Gx36CQ-M
- BR-SLX9850-10Gx72S-D
- BR-SLX9850-100Gx36CQ-D
- BR-SLX9850-100Gx12CQ-M

TABLE 1 SLX 9850 interface modules capabilities

Capability	Modular interface module
MPLS	Yes
Packet buffer memory per interface module	12GB (BR-SLX9850-10Gx72S-M) 36GB (BR-SLX9850-100Gx36CQ-M) 8GB (BR-SLX9850-10Gx72S-D) 24GB (BR-SLX9850-100Gx36CQ-D) 8GB (BR-SLX9850-100Gx12CQ-M)

What's new in this document

On October 30, 2017, Extreme Networks, Inc. acquired the data center networking business from Brocade Communications Systems, Inc. This document has been updated to remove or replace references to Brocade Communications, Inc. with Extreme Networks., Inc., as appropriate.

This document is released with Extreme SLX OS 17r.2.01.

The following table includes descriptions of new information added to this guide for the SLX OS 17r.2.00 software release.

TABLE 2 Summary of enhancements in SLX OS release 17r.2.00

Feature	Description	Described in
MPLS transit load balancing	MPLS transit load balancing provides support for SLX-OS MPLS packet load balancing based on the inner headers when the Layer 2 optimized TCAM profile configuration is activated.	Configuration considerations for enabling MPLS on a LAG interface on page 31

For complete release information, refer to the SLX OS Release Notes.

MPLS Traffic Engineering

MPLS Traffic Engineering overview

Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) is supported on Extreme devices. MPLS can be used to direct packets through a network over a pre-determined path of routers. Forwarding decisions in MPLS are based on the contents of a label applied to the packet.

Traffic engineering is the ability to direct packets through a network efficiently, using information gathered about network resources. When used as an application of MPLS, traffic engineering involves creating paths that make the best use of available network resources, avoiding points of congestion and making efficient use of high bandwidth interfaces. Packets traveling over these paths are forwarded using MPLS.

IETF RFC and Internet draft support

The implementation of MPLS supports the following IETF RFCs and Internet Drafts.

MPLS

RFC 3031 - Multiprotocol Label Switching Architecture.

RFC 3032 - MPLS Label Stack Encoding.

RFC 3036 - LDP Specification.

RFC 2205 - Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification.

RFC 2209 - Resource ReSerVation Protocol (RSVP) -- Version 1 Message Processing Rule.

RFC 3209 - RSVP-TE.

RFC 3270 - MPLS Support of Differentiated Services.

RFC 4090 - Facility backup and Fast Reroute.

OSPF

RFC 3630 Traffic Engineering (TE) Extensions to OSPF v2.

IS-IS

RFC 3784 Intermediate System to Intermediate System (IS-IS) Extensions for Traffic Engineering (TE).

How MPLS works

MPLS uses a *label switching* forwarding method to direct packets through a network. In label switching, a packet is assigned a label and passes along a predetermined path of routers. Forwarding decisions are based on the contents of the label, rather than information in the packet's IP header.

The following sections describe these basic MPLS concepts:

- How packets are forwarded through an MPLS domain
- The kinds of *Label Switched Paths (LSPs)* that can be configured on a device
- The components of an MPLS label header

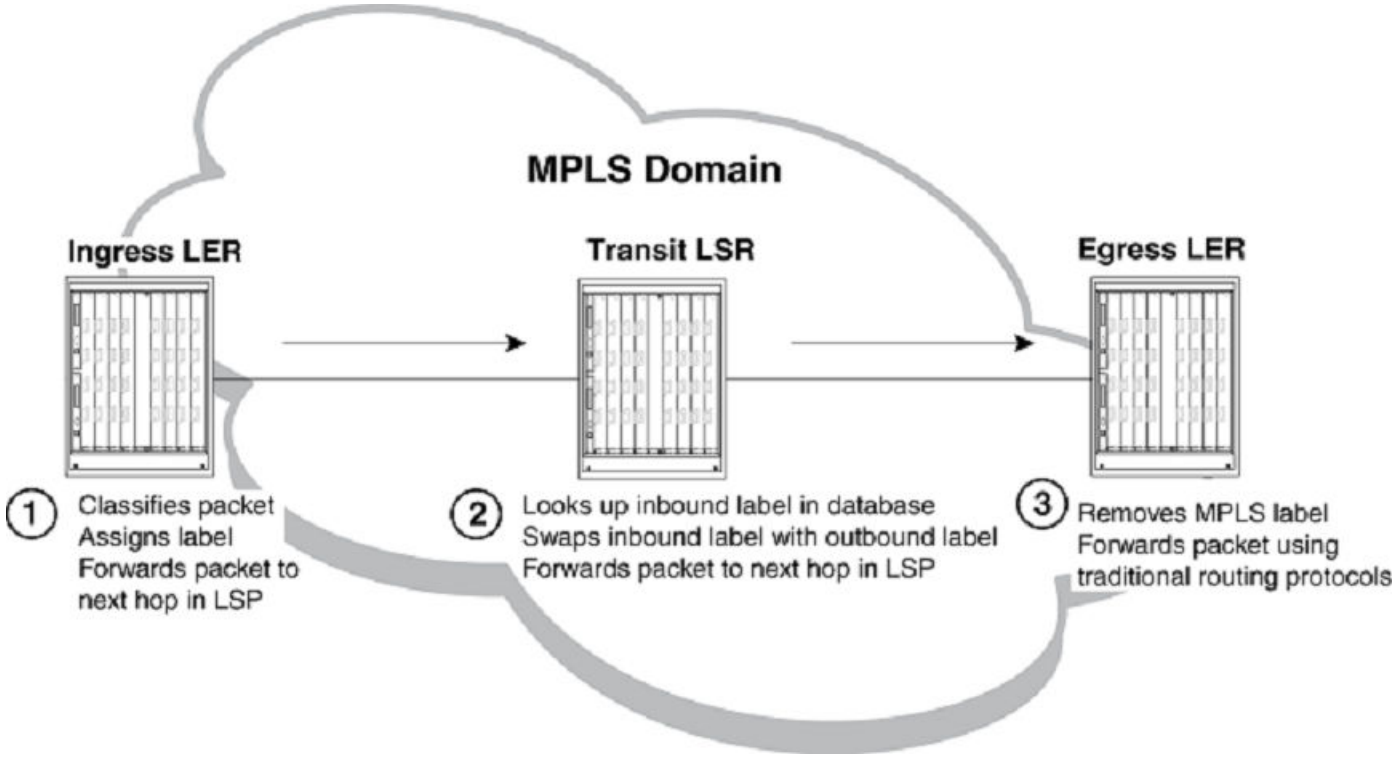
How packets are forwarded through an MPLS domain

An *MPLS domain* consists of a group of MPLS-enabled routers, called *Label Switching Routers (LSRs)*. In an MPLS domain, packets are forwarded from one MPLS-enabled router to another along a predetermined path, called an LSP. LSPs are one-way paths between MPLS-enabled routers on a network. To provide two-way traffic, the user configures LSPs in each direction.

The LSRs at the headend and tailend of an LSP are known as *Label Edge Routers (LERs)*. The LER at the headend, where packets enter the LSP, is known as the *ingress LER* . The LER at the tailend, where packets exit the LSP, is known as the *egress LER* . Each LSP has one ingress LER and one egress LER. Packets in an LSP flow in one direction: from the ingress LER towards the egress LER. In between the ingress and egress LERs there may be zero or more *transit LSRs* . A device enabled for MPLS can perform the role of ingress LER, transit LSR, or egress LER in an LSP. Further, a device can serve simultaneously as an ingress LER for one LSP, transit LSR for another LSP, and egress LER for some other LSP.

Label switching in an MPLS domain depicts an MPLS domain with a single LSP consisting of three LSRs: an ingress LER, a transit LSR, and an egress LER.

FIGURE 1 Label switching in an MPLS domain



Label switching in an MPLS domain works as described below.

1. The Ingress LER receives a packet and pushes a label onto it.

   When a packet arrives on an MPLS-enabled interface, the device determines to which LSP (if any) the packet are assigned. Specifically, the device determines to which *Forwarding Equivalence Class (FEC)* the packet belongs. An FEC is simply a group of packets that are all forwarded in the same way. For example, a FEC could be defined as all packets from a given *Virtual Leased Line (VLL)*. FECs are mapped to LSPs. When a packet belongs to a FEC, and an LSP is mapped to that FEC, the packet is assigned to the LSP.
   When a packet is assigned to an LSP, the device, acting as an ingress LER, applies (pushes) a tunnel label onto the packet. A label is a 32-bit, fixed-length identifier that is significant only to MPLS. Refer to MPLS label header encoding on page 21 for specific information about the contents of a label. From this point until the packet reaches the egress LER at the end of the path, the packet is forwarded using information in its label, not information in its IP header. The packet's IP header is not examined again as long as the packet traverses the LSP. The ingress LER may also apply a VC label onto the packet based on the VPN application.

   On the ingress LER, the label is associated with an outbound interface. After receiving a label, the packet is forwarded over the outbound interface to the next router in the LSP.

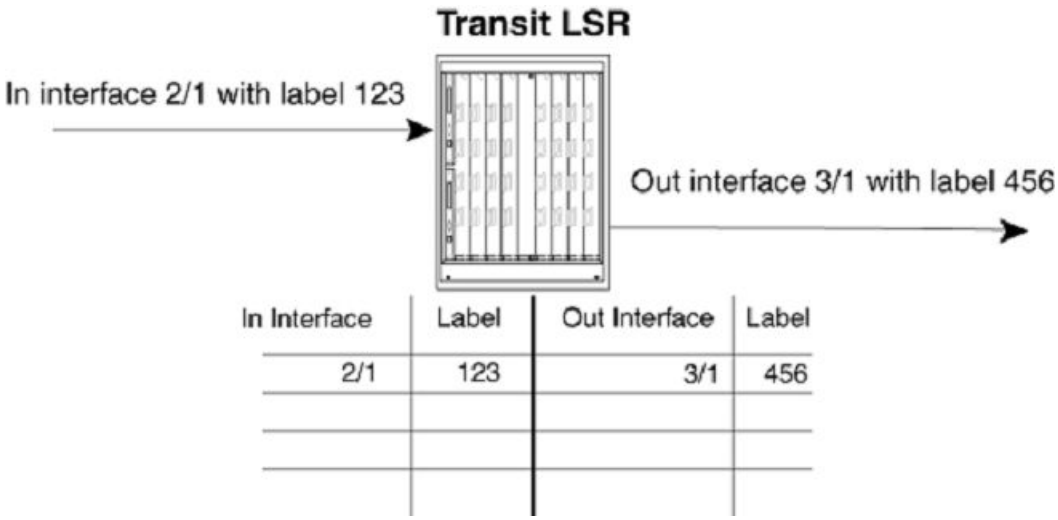
2. A transit LSR receives the labeled packet, swaps the label, and forwards the packet to the next LSR.

   In an LSP, zero or more transit LSRs can exist between the ingress and egress LERs. A transit LSR swaps labels on an MPLS packet and forwards the packet to the next router in the LSP.

   When a transit LSR receives an MPLS packet, it looks up the label in its *MPLS forwarding table* . This table maps the label and inbound interface to a new label and outbound interface. The transit LSR replaces the old label with the new label and sends the packet out the outbound interface specified in the table. This process repeats at each transit LSR until the packet reaches the next-to-last LSR in the LSP (for signaled LSPs).

   Figure 2 illustrates an example of the label swapping process on a transit LSR.

   FIGURE 2 Label swapping on a transit LSR



   In this example, a packet comes into interface 2/1 with label 123. The transit LSR then looks up this interface-label pair in its MPLS forwarding table. The inbound interface-label pair maps to an outbound-interface-label pair - in this example, interface 3/1 with label 456. The LSR swaps label 123 with label 456 and forwards the packet out interface 3/1.

3. The egress LER receives labeled packet, pops label, and forwards IP packet.

   When the packet reaches the egress LER, the MPLS label is removed (called *popping* the label), and the packet can then be forwarded to its destination using standard hop-by-hop routing protocols. On signaled LSPs, the label is popped at the penultimate (next to last) LSR, rather than the egress LER. Refer to Penultimate hop popping on page 23 for more information.

MPLS label header encoding

The following diagram illustrates the structure of the 32-bit MPLS label header. When a packet enters an LSP, the ingress LER pushes a label onto the packet.

FIGURE 3 Structure of an MPLS Label Header

An MPLS label header is composed of the following parts:

Label value (20 bits)

The label value is an integer in the range 16 - 1048575. (Labels 0 - 15 are reserved by the IETF for special usage.) For signaled LSPs, the device dynamically assigns labels in the range 1024 - 499999.

EXP field (3 bits)

The EXP field is designated for experimental usage. By default, a device uses the EXP field to define a Class of Service (CoS) value for prioritizing packets traveling through an LSP. Please refer to MPLS Traffic Engineering on page 17, for more information. Note that software forwarded VPLS packets do not use the EXP encode table.

S (Bottom of Stack) field (one bit)

An MPLS packet can be assigned multiple labels. When an MPLS packet has multiple labels, they are logically organized in a last-in, first-out *label stack*. An LSR performs a pop or swap operation on the topmost label; that is, the most recently applied label in the stack. The Bottom of Stack field indicates whether this label is the last (oldest) label in the stack. When the label is the last one in the stack, the Bottom of Stack field is set to one. If not, the Bottom of Stack field is set to zero.

A device acting as an LSR can perform one push, swap, or pop operation on an incoming MPLS packet. The device can accept MPLS packets that contain multiple labels, but only the topmost label is acted upon.

TTL field (eight bits)

The TTL field indicates the *Time To Live (TTL)* value for the MPLS packet. At the ingress LER, an IP packet's TTL value is copied to its MPLS TTL field. At each transit LSR hop, the MPLS TTL value is decremented by one. When the MPLS TTL value reaches zero, the packet is discarded. Optionally, the user can configure the LSRs not to decrement the MPLS TTL value at each hop.

Using MPLS in traffic engineering

Traffic engineering is the task of routing network traffic to avoid points of congestion and make efficient use of high bandwidth interfaces. When used as an application of MPLS, traffic engineering involves creating LSPs that make the best use of available network resources; that is, *traffic-engineered LSPs*. This section explains the process of creating traffic-engineered LSPs.

Creating traffic-engineered LSPs involves the following tasks:

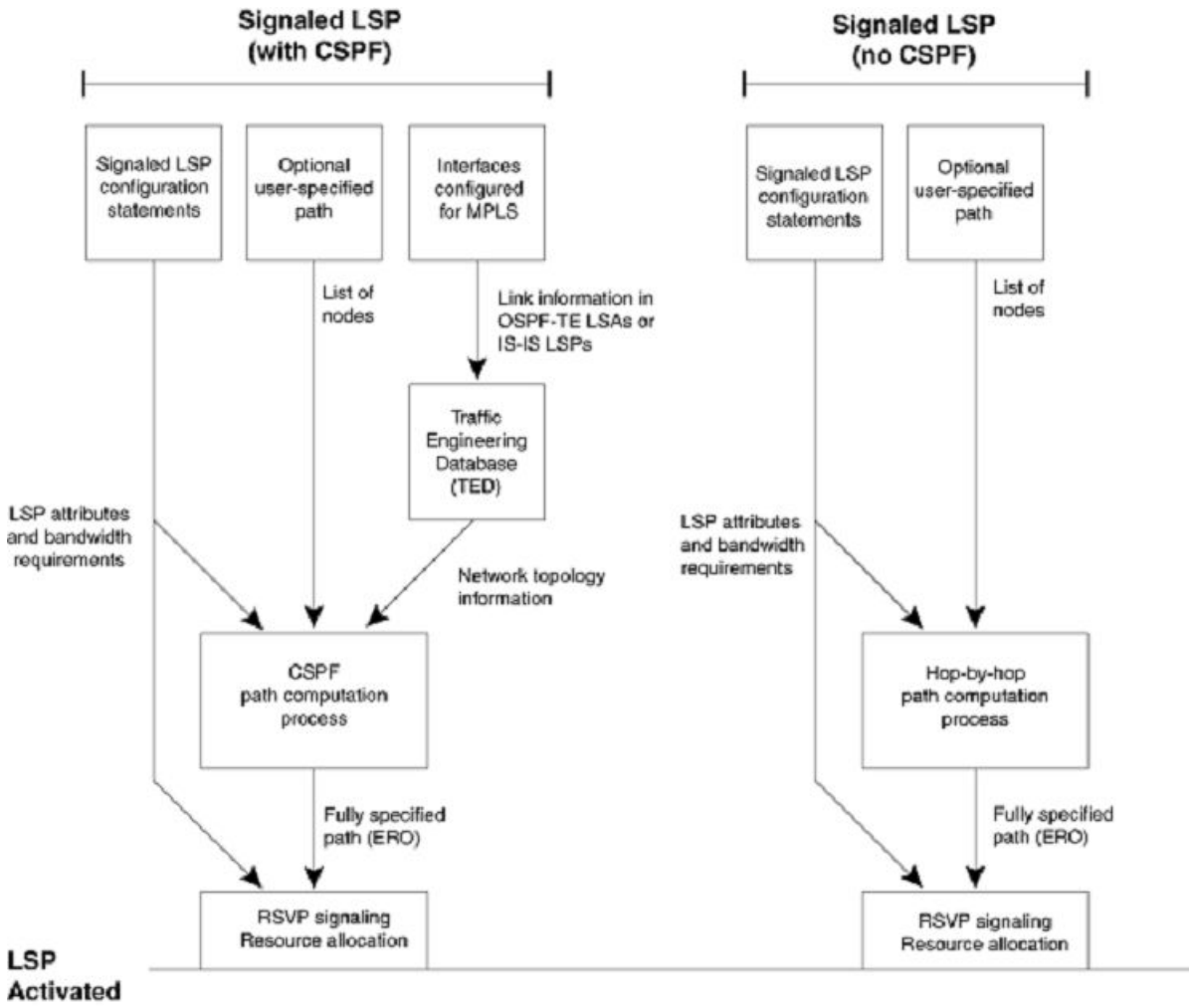
- Gathering information about the network

- Using the gathered information to select optimal paths through the network
- Setting up and maintaining the paths

For traffic-engineered signaled LSPs, devices can perform these tasks dynamically. Figure 4 illustrates the process that takes place to configure, establish, and activate traffic-engineered signaled LSPs.

NOTE
Adaptive LSPs can have primary and secondary sessions up at the same time. Extreme devices support a maximum of 5000 LSPs, and a maximum of 10000 sessions.

FIGURE 4 How traffic-engineered LSPs are configured, established, and activated



CSPF calculates a traffic-engineered path

When the user configures a signaled Label Switched Path, the user specifies the address of the egress LER, as well as optional attributes, such as the LSPs priority and bandwidth requirements. The user can optionally specify a path of LSRs that the LSP must pass through on the way to the egress LER. When the user enables the signaled LSP, the *Constrained Shortest Path First (CSPF)* process on the ingress LER uses this information to calculate a *traffic-engineered path* between the ingress and egress LERs.

CSPF is an advanced form of the *Shortest Path First (SPF)* process used by IGP routing protocols. The CSPF process on the ingress LER uses the configured attributes of the LSP, user-specified path (when there is one), and the information in the *Traffic Engineering Database (TED)* to calculate the traffic-engineered path. This process consists of a sequential list of the physical interfaces that packets assigned to this LSP pass through to travel from the ingress LER to the egress LER. The traffic-engineered path takes into account the network topology, available resources, and user-specified constraints. The traffic-engineered path calculated by CSPF may or may not be the same as the shortest path that would normally be calculated by standard IGP routing protocols.

CSPF is enabled by default for signaled LSPs, but can be disabled. When signaled LSPs are configured without CSPF, the shortest path from the ingress LER to the egress LER is calculated using standard hop-by-hop routing methods. When the LSP also is configured to use a user-specified path, the device calculates the shortest path between each LSR in the path. As with CSPF, the output of this process is a fully specified path of physical interfaces on LSRs.

The advantage of configuring signaled LSPs without CSPF is that it can span multiple IS-IS levels. Since IS-IS LSPs with TE extensions have an area and level flooding scope, the information in an LSRs TED is relevant only to their area or level. Consequently, signaled LSPs that use CSPF can span only an IS-IS level. Signaled LSPs that do not use CSPF, because they do not rely on information in the TED, do not have this restriction.

Once the path for the LSP has been calculated, RSVP signaling then causes resources to be reserved and labels to be allocated on each LSR specified in the path. This may cause already existing, lower priority LSPs to be preempted. Once resources are reserved on all the LSRs in the path, the signaled LSP is considered to be activated; that is, packets can be forwarded over it.

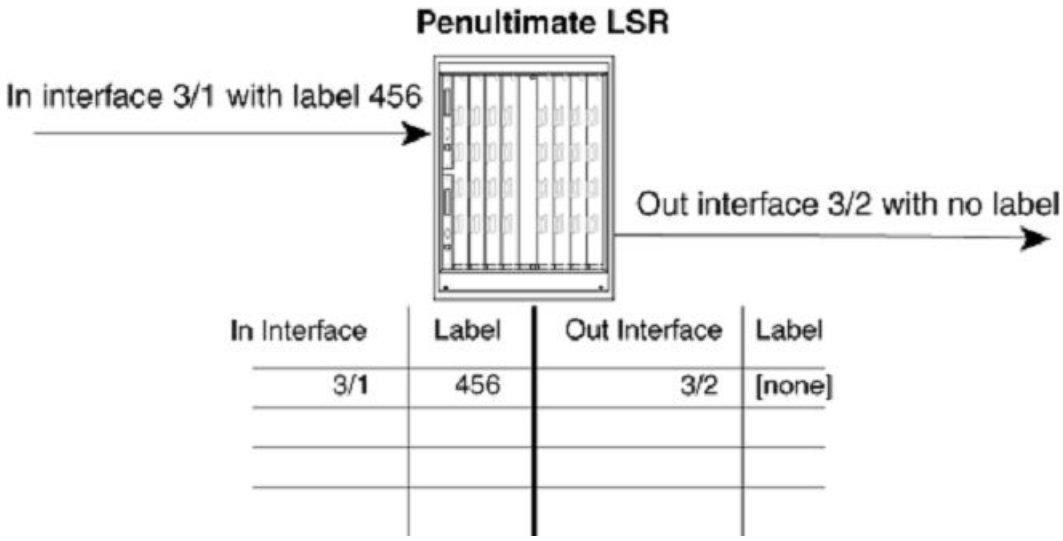
The following sections provide additional information about the individual components of the process for activating traffic-engineered signaled LSPs, illustrated in Using MPLS in traffic engineering on page 21.

Penultimate hop popping

On signaled LSPs, the MPLS label is popped at the next-to-last LSR in the LSP, instead of at the egress LER. This action is called *penultimate hop popping*. Penultimate hop popping improves forwarding efficiency by allowing the egress LER to avoid performing both a MPLS forwarding table lookup and an IP forwarding table lookup for each packet exiting the LSP. Instead, the MPLS label is popped at the penultimate (next-to-last) LSR, and the packet is forwarded to the egress LER with no MPLS encoding. The egress LER, in fact, does not recognize the packet as emerging from an LSP.

Figure 5 illustrates the operation that takes place at the penultimate LSR in an LSP.

FIGURE 5 Penultimate hop popping



When an LSR receives an MPLS packet, it looks up the label in its MPLS forwarding table. Normally, this table maps the label and inbound interface to a new label and outbound interface. However, when this is the penultimate LSR in an LSP, the label and inbound interface map only to an outbound interface. The penultimate LSR pops the label and forwards the packet - now a regular IP packet - out the outbound interface. When the packet reaches the egress LER, there is no indication that it had been forwarded over an LSP. The packet is forwarded using standard hop-by-hop routing protocols.

MPLS CSPF fate-sharing group

A MPLS CSPF fate-sharing group or a *Shared Risk Link Group (SRLG)* is a method used to group *Traffic Engineering (TE)* links and nodes in a network that share the same risk of failure. The user can influence the path computation for a CSPF-enabled LSP by configuring a CSPF fate-sharing group so that both the protected path and the backup path avoid sharing the same TE links traversed. The path computation for a CSPF-enabled LSP uses the information from the TE database to compute the best path for an LSP satisfying all constraints (bandwidth reservations, network topology information, available resources), yet has the shortest distance to its destination. The CSPF computation for an LSP only uses the information from the TE database at the time of computation. Any future updates to the TE database do not cause the CSPF-enabled LSP to recompute. Each CSPF fate-sharing group has an associated penalty (or cost) assigned to it. The penalty associated with a CSPF fate-sharing group is used to direct the path computation for a CSPF-enabled LSP away from TE links that share the same risk used by the set of TE links that the protected path is using. The greater the penalty associated with a group, the less likely the secondary shares TE links used by the protected path.

A CSPF fate-sharing group is identified by a group name, and uses the following four ways to identify elements in the TE database:

- Interface address - The interface address identifies all TE links by either the local address, or the remote address matching the configured interface address.
- Point-to-point link - A point-to-point link identifies TE links by the local address and the remote address on an interface. A point-to-point link specifies the *from* address and the *to* address. The order in which the address is configured is not significant.
- Node - The node address is used to identify the device. All TE links from this device are included.
- Subnet - The IP address with subnet mask identifies all TE links by either the local interface or the remote address belonging to the configured subnet.

A CSPF fate-sharing group can be used for setting up a secondary LSP when the associated primary LSP is in an UP state.

Refer to, Configuring an MPLS CSPF fate-sharing group on page 25 for more information on configuring the path computation for a CSPF-enabled LSP using CSPF fate-sharing group information.

Configuration considerations when using CSPF fate-sharing group information

Consider the following when using CSPF fate-sharing group information:

> NOTE
> This release only supports a single mode of CSPF computation for a CSPF group by adding penalties to each TE link's native IGP cost when it shares fate-sharing groups used by the protected path.

- CSPF computation using a CSPF group is only applicable when computing a secondary LSP path. It is not applicable to the primary or protected LSP path.
- CSPF calculates the least cost paths first and then applies the hop limit on the paths.
- CSPF computation using a CSPF group is used only for computing the secondary LSP path when the primary LSP is in an UP state. In this case, CSPF collects group information from all TE links used by the primary LSP. For each TE link, CSPF computes the total adjusted distance. The total adjusted distance for each TE link is equal to the native IGP cost of the TE link

plus the sum of all penalties of the CSPF groups that the TE link is associated with, and used by the primary LSP. For example, Q1, Q2, and Q3 is a collection of CSPF groups used by the primary LSP. TE link 1 is a member of CSPF groups Q1 and Q2. Q1 has a penalty of 10, and Q2 has a penalty 30. The total penalty of CSPF groups Q1 and Q2 is equal to 40. The total adjusted distance for TE link 1 is equal to the native IGP cost plus 40. The penalty is only applied once to each shared CSPF group that the TE link is associated with. The secondary LSP path is then computed from ingress to egress using the adjusted distance of each TE link.

Configuring an MPLS CSPF fate-sharing group

To configure a CSPF fate-sharing group, perform the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable the policy mode.

```
device(config-router-mpls)# policy
```

4. Specify the CSPF group computation mode for a fate-sharing group, and enable the **penalty** option by entering the following command in MPLS policy mode.

```
device(config-router-mpls-policy)# cspf-group computation-mode add-penalty
```

5. Configure a CSPF fate-sharing group by assigning a name to the group. Enter the following command in router MPLS mode.

```
device(config-router-mpls)# cspf-group group3
```

6. Set the penalty value for the CSPF fate-sharing group. Enter the following command.

```
device(config-router-mpls-cspf-group-group3)# penalty 100
```

7. Configure the local address of the CSPF fate-sharing group. Enter the following command.

```
device(config-router-mpls-cspf-group-group3)# from 10.1.1.1
```

8. Configure the local address and remote address on a point-to-point link of the CSPF fate-sharing group, enter the following command.

```
device(config-router-mpls-cspf-group-group3)# link 10.1.1.1  10.1.1.2
```

9. Configure the local Subnet IP address with the subnet mask length. Enter the following command.

```
device(config-router-mpls-cspf-group-group3)# subnet 10.1.2.0/24
```

10. To penalize all links from the node IP address, enter the following command.

```
device(config-router-mpls-cspf-group-group3)# node 10.1.1.1
```

The following example configures a CSPF fate-sharing group.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# cspf-group computation-mode add-penalty
device(config-router-mpls)# cspf-group group3
device(config-router-mpls-cspf-group-group3)# penalty 100
device(config-router-mpls-cspf-group-group3)# from 10.1.1.1
device(config-router-mpls-cspf-group-group3)# link 10.1.1.1  10.1.1.2
device(config-router-mpls-cspf-group-group3)# subnet 10.1.2.0/24
device(config-router-mpls-cspf-group-group3)# node 10.1.1.1
```

Deleting CSPF groups

Deleting a CSPF group enables users to delete all the CSPF fate-share groups using a single command on all Extreme devices running MPLS. Users are required to confirm execution with a warning message.

In this example, group3 has already been set up as a fate-sharing CSPF group. To delete this CSPF fate-sharing group, complete the following command in router MPLS mode.

```
device(config-router-mpls)# no cspf-group group3
```

The *group-name* variable (in this example, "group3"), specifies the name of the fate-sharing group and can be up to 128 characters. The objects that can be specified for a fate-sharing group are interface, point-to-point link, node, and subnet. The maximum number of CSPF fate-sharing groups that can be configured on a device is 1000. To delete each configuration group individually, enter the above command with the relevant value for the group-name argument.

The user can delete all configured groups at once. Use a single **no cspf-group** command. This command is only available at the router-mpls level and takes no arguments.

Deleting a CSPF group sample configuration

```
device# configure
device(config)# router mpls
device(config-router-mpls)# no cspf-group group3
```

At this point, all of the CSPF groups are deleted at once.

Displaying CSPF fate-sharing group configuration

To display CSPF fate-sharing group configuration for all groups configured on a device, use the **show mpls configuration** command or the **show run** command.

To display CSPF fate-sharing group information for a specific CSPF group, use the **show show running-config router mpls cspf-group** *cspf-group name* command. The output from the **show running-config router mpls** command. In the following example output, the CSPF fate-sharing group information is displayed for CSPF group *gold* .

```
device# show running-config router mpls cspf-group gold
cspf-group gold
penalty 65535
node 10.7.7.3
node 10.7.7.8
```

Fate-sharing group membership for any given TE link or node consists of its own membership to the group, and the TE node to which it belongs. The output from the **show mpls te database detail** command is enhanced to display the fate-sharing groups to which the TE

links or nodes belong. In the following example output, node 10.20.20.20 displays fate-sharing group information for group1/100 and group2/10.

```
TE Router ID: 30.11.1.30
 OSPF Area: 0.0.0.0
  Node Id: (14.1.2.3), Type: Router
   P2P Link: From: 14.1.2.3 To: 13.13.13.13, Local: 13.14.1.14, Remote: 13.14.1.13, LSA Id: 16777313, Gen:
927
    Admin Group: 0x0
    IGP Metric: 1
    TE Metric: 1
    Link BW: 1000000 kbits/sec
    Reservable BW: 1000000 kbits/sec
    Unreserved BW:
     [0]    1000000 kbits/sec   [1]    1000000 kbits/sec
     [2]    1000000 kbits/sec   [3]    1000000 kbits/sec
     [4]    1000000 kbits/sec   [5]    1000000 kbits/sec
     [6]    1000000 kbits/sec   [7]    1000000 kbits/sec
   P2P Link: From: 14.1.2.3 To: 13.13.13.13, Local: 13.14.20.14, Remote: 13.14.20.13, LSA Id: 16777314, Gen:
916
    Admin Group: 0x0
    IGP Metric: 1
    TE Metric: 1
    Link BW: 1000000 kbits/sec
    Reservable BW: 1000000 kbits/sec
    Unreserved BW:
     [0]    1000000 kbits/sec   [1]    1000000 kbits/sec
     [2]    1000000 kbits/sec   [3]    1000000 kbits/sec
     [4]    1000000 kbits/sec   [5]    1000000 kbits/sec
     [6]    1000000 kbits/sec   [7]    1000000 kbits/sec
   P2P Link: From: 14.1.2.3 To: 19.19.19.19, Local: 19.14.1.14, Remote: 19.14.1.19, LSA Id: 16777223, Gen:
934
    Admin Group: 0x0
    IGP Metric: 1
    TE Metric: 1
    Link BW: 10000000 kbits/sec
    Reservable BW: 10000000 kbits/sec
    Unreserved BW:
     [0]   10000000 kbits/sec   [1]   10000000 kbits/sec
     [2]   10000000 kbits/sec   [3]   10000000 kbits/sec
     [4]   10000000 kbits/sec   [5]   10000000 kbits/sec
     [6]   10000000 kbits/sec   [7]   10000000 kbits/sec
   P2P Link: From: 14.1.2.3 To: 57.57.57.57, Local: 14.57.1.14, Remote: 14.57.1.57, LSA Id: 16777241, Gen:
886
    Admin Group: 0x0
    IGP Metric: 1
    TE Metric: 1
    Link BW: 10000000 kbits/sec
    Reservable BW: 10000000 kbits/sec
    Unreserved BW:
     [0]   10000000 kbits/sec   [1]   10000000 kbits/sec
     [2]   10000000 kbits/sec   [3]   10000000 kbits/sec
     [4]   10000000 kbits/sec   [5]   10000000 kbits/sec
     [6]   10000000 kbits/sec   [7]   10000000 kbits/sec
  Node Id: (4.4.4.4), Type: Router
   P2P Link: From: 4.4.4.4 To: 17.17.17.17, Local: 23.45.67.8, Remote: 23.45.67.17, LSA Id: 16777220, Gen:
938
    Admin Group: 0x0
    IGP Metric: 1
    TE Metric: 1
    Link BW: 1000000 kbits/sec
    Reservable BW: 1000000 kbits/sec
    Unreserved BW:
     [0]    1000000 kbits/sec   [1]    1000000 kbits/sec
     [2]    1000000 kbits/sec   [3]    1000000 kbits/sec
     [4]    1000000 kbits/sec   [5]    1000000 kbits/sec
     [6]    1000000 kbits/sec   [7]    1000000 kbits/sec
  Node Id: (13.13.13.13), Type: Router
   P2P Link: From: 13.13.13.13 To: 14.1.2.3, Local: 13.14.1.13, Remote: 13.14.1.14, LSA Id: 16777218, Gen:
928
    Admin Group: 0x0
```

```
    IGP Metric: 1
    TE Metric: 1
    Link BW: 1000000 kbits/sec
    Reservable BW: 1000000 kbits/sec
    Unreserved BW:
     [0]    1000000 kbits/sec   [1]    1000000 kbits/sec
     [2]    1000000 kbits/sec   [3]    1000000 kbits/sec
     [4]    1000000 kbits/sec   [5]    1000000 kbits/sec
     [6]    1000000 kbits/sec   [7]    1000000 kbits/sec
   P2P Link: From: 13.13.13.13 To: 14.1.2.3, Local: 13.14.20.13, Remote: 13.14.20.14, LSA Id: 16777219, Gen:
915
    Admin Group: 0x0
    IGP Metric: 1
    TE Metric: 1
    Link BW: 1000000 kbits/sec
    Reservable BW: 1000000 kbits/sec
    Unreserved BW:
     [0]    1000000 kbits/sec   [1]    1000000 kbits/sec
     [2]    1000000 kbits/sec   [3]    1000000 kbits/sec
     [4]    1000000 kbits/sec   [5]    1000000 kbits/sec
     [6]    1000000 kbits/sec   [7]    1000000 kbits/sec
   P2P Link: From: 13.13.13.13 To: 11.12.13.14, Local: 11.13.1.13, Remote: 11.13.1.11, LSA Id: 16777220,
Gen:919
    Admin Group: 0x0
    IGP Metric: 1
    TE Metric: 1
    Link BW: 1000000 kbits/sec
    Reservable BW: 1000000 kbits/sec
    Unreserved BW:
     [0]    1000000 kbits/sec   [1]    1000000 kbits/sec
     [2]    1000000 kbits/sec   [3]    1000000 kbits/sec
     [4]    1000000 kbits/sec   [5]    1000000 kbits/sec
     [6]    1000000 kbits/sec   [7]    1000000 kbits/sec
   P2P Link: From: 13.13.13.13 To: 11.12.13.14, Local: 11.13.2.13, Remote: 11.13.2.11, LSA Id: 16777221,
Gen:921
    Admin Group: 0x0
    IGP Metric: 1
    TE Metric: 1
    Link BW: 1000000 kbits/sec
    Reservable BW: 1000000 kbits/sec
    Unreserved BW:
     [0]    1000000 kbits/sec   [1]    1000000 kbits/sec
     [2]    1000000 kbits/sec   [3]    1000000 kbits/sec
     [4]    1000000 kbits/sec   [5]    1000000 kbits/sec
     [6]    1000000 kbits/sec   [7]    1000000 kbits/sec
```

The **show running-config router mpls lsp** *lsp_name* command displays detailed information about a specific LSP name. The output from the **show running-config router mpls lsp** *lsp_name* command is enhanced to display whether the fate-sharing group information is applied to the path computation for a specified LSP. When the fate-sharing group information is applied, "yes" is displayed in the field. When fate-sharing group information is not applied, "no" is displayed in the field. Fate-sharing group information can also be applied to the path computation for a secondary LSP or a bypass LSP path. In the following example, the fate-sharing group information is applied to LSP test2.

```
device# show running-config router mpls lsp name test2
LSP test2, to 10.100.100.100
 From: 10.20.20.20, admin: UP, status: UP, tunnel interface(primary path): tnl3
 Times primary LSP goes up since enabled: 1
 Metric: 0, number of installed aliases: 0 Adaptive
 Maximum retries: NONE, no. of retries: 0
 Pri. path: NONE, up: yes, active: yes
 Setup priority: 7, hold priority: 0
 Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
 Constraint-based routing enabled: yes
 Tie breaking: random, hop limit: 0
 LDP tunneling enabled: no
 Sec. path: path2, active: no
  Hot-standby: yes, status: up
  Setup priority: 7, hold priority: 0
  Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
```

```
  Constraint-based routing enabled: yes
  Fate-sharing group applied: yes
  hop limit: 0
 Active Path attributes:
  Tunnel interface: tnl3, outbound interface: e1/1
  Tunnel index: 2, Tunnel instance: 1 outbound label: 3
  Path calculated using constraint-based routing: yes
  Path calculated using interface constraint: no
  Recorded routes:
   Protection codes: P: Local N: Node B: Bandwidth I: InUse
   10.1.1.1
```

In the following example output, the primary LSP path, and the secondary bypass LSP path is UP. The **add-penalty** parameter is enabled under the CSPF group computation mode as highlighted below.

```
device# show running-config router mpls lsp name test2
LSP test2, to 10.100.100.100
 From: 10.20.20.20, admin: UP, status: UP, tunnel interface(primary path): tnl3
 Times primary LSP goes up since enabled: 1
 Metric: 0, number of installed aliases: 0 Adaptive
 Maximum retries: NONE, no. of retries: 0
 Pri. path: NONE, up: yes, active: yes
 ...
Sec. path: path2, active: no
 Hot-standby: yes, status: up
  Setup priority: 7, hold priority: 0
  Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
  Constraint-based routing enabled: yes
 Path cspf-group computation-mode: add-penalty
```

IS-IS Link State Protocol data units with TE extensions for MPLS interfaces

An MPLS-enabled device running IS-IS can be configured to send out *Link State Protocol (LSP)* data units that contain special extensions to support *Traffic Engineering (TE)*. (In this section -- and nowhere else in this chapter -- LSP is the acronym for Link State Protocol. In other sections, LSP means Label Switched Path.) These LSPs are composed of a fixed header and a number of tuples known as *Type/Length/Value triplets (TLVs)*. LSPs that are used for traffic engineering contain a new object called a sub-TLV. Sub-TLVs are similar to regular TLVs except that, where regular TLVs exist inside IS-IS packets, sub-TLVs reside within regular TLVs. Each sub-TLV consists of three fields: a one-octet Type field, a one-octet Length field, and zero or more octets of Value.

These LSPs are flooded throughout the IS-IS domain. LSRs that receive the IS-IS LSPs with TE extensions place the traffic engineering information into a *Traffic Engineering Database (TED)*, which maintains topology data about the nodes and links in the MPLS domain.

IS-IS LSPs have special extensions that contain information related to traffic engineering and are described in *RFC 3784*. The extensions consist of Type/Length/Value triplets (sub-TLVs) containing the following information:

- IP address of the local interface for the link
- IP address of the remote interface for the link (for point-to-point adjacencies)
- Traffic engineering metric for the link (by default, this is equal to the IS-IS link cost)
- Maximum bandwidth on the interface
- Maximum reservable bandwidth on the interface
- Unreserved bandwidth on the interface
- Administrative groups to which the interface belongs

When configured to do so, the device sends out IS-IS LSPs with TE extensions for each of its MPLS-enabled interfaces. The user can optionally specify the maximum amount of bandwidth that can be reserved on an interface, as well as assign interfaces to administrative groups. Refer to Setting traffic engineering parameters for MPLS interfaces on page 36 for more information.

Any of the following events trigger the device to send out IS-IS LSPs with a TE extension:

- Change in the interface's administrative group membership.
- Change in the interface's maximum available bandwidth or maximum reservable bandwidth.
- Significant change in unreserved bandwidth per priority level, which can be either of the following:
  - For any priority level, the difference between the previously advertised, unreserved bandwidth and the current, unreserved bandwidth exceeds five percent of the maximum reservable bandwidth.
  - Any change when the total reserved bandwidth exceeds 95 percent of the maximum reservable bandwidth.

In addition, IS-IS LSPs with TE extensions can be triggered by IS-IS (for example, when an interface's link state changes). Furthermore, when an interface is no longer enabled for MPLS, the device stops sending out IS-IS LSPs with TE extensions for that interface.

Configuring MPLS

Enabling MPLS

MPLS is disabled by default. To enable MPLS on a device, the user must perform the steps listed below.

1. Enable MPLS on the device
2. Enable MPLS on individual interfaces
3. Set global MPLS policy parameters (optional)
4. Set traffic engineering parameters for MPLS-enabled interfaces (optional)
5. Set RSVP parameters (optional)

Enabling MPLS on the device

To enable MPLS on the device, enter the following commands.

```
device# configure
device(config)# router mpls
```

To disable MPLS on the device, use the [**no**] form of the command.

Enabling MPLS on individual interfaces

After the user enables MPLS globally on the device, the user can enable it on one or more interfaces. For example, to enable MPLS on interface ethernet *3/1* .

```
device(config-router-mpls)# mpls-interface ethernet 3/1
```

MPLS over virtual Ethernet interfaces

MPLS over VE interfaces enables MPLS to be configured over tagged links.

Extreme devices support MPLS over virtual ethernet (VE) interfaces. MPLS can run over a single tag on the port. Other tags on the port can be used for other applications, such as Layer 2 VLANs, VPLS endpoints, and VLL endpoints.

An MPLS enabled VE interface supports the following services.

- Internet Protocol over Multi-Protocol Label Switching (IP over MPLS)
- Transit Label Switching Router (LSR)
- Policy Based Routing (PBR) over MPLS
- Label-switched Path (LSP) Accounting
- MPLS Virtual leased Line (VLL)
- MPLS Virtual Private LAN Service (VPLS)
- 802.1ag
- MPLS Operations, Administration, and Management (OAM)

Configuration considerations before enabling MPLS on a VE interface

Before enabling MPLS on a VE interface, consider the configuration notes in this section.

- The user must create a VE *vid* virtual interface ID. The virtual interface ID is a decimal number that represents an already configured VE interface.
- At least one IP address must be configured over a VE interface.
- The user can enable MPLS on two or more tags on the same port.
- In the output of the **show vlan** command, MPLS packets that are received on an MPLS enabled VE interface are displayed in the Bytes received field.

Configuration considerations for enabling MPLS on a LAG interface

When MPLS is globally enabled on the device, a port that is configured in a LAG can be enabled as an MPLS interface port to create an MPLS LAG. The user can do this through either of the following approaches:

- Include a primary LAG port that has already been MPLS-enabled in a new LAG
- MPLS-enable a primary LAG port of a previously configured LAG

The user must consider the following points when configuring MPLS on a LAG:

- MPLS configuration on dynamic lag interfaces are supported
- Switch and LACP LAGs are not supported
- MPLS is enabled on the primary port of the LAG and this enables MPLS on the entire LAG. Secondary ports of the LAG cannot be individually configured for MPLS.

NOTE

A hashing scheme for enhanced load balancing allows MPLS transit load balancing to include inner headers of different packet types in parallel. This hashing scheme is supported only with the "Layer 2 Optimized" Tcam profile and is enabled by default when the **profile tcam layer2-optimised-1** configuration is activated. The "Error: Operation not supported in the current hardware TCAM profile" message is displayed for the following commands because these functionalities are already taken care of by this enhanced hashing scheme:

- lag hash speculate-mpls inner-eth
- lag hash speculate-mpls inner-ip-raw
- lag hash speculate-mpls inner-ip-tag
- lag hash speculate-mpls inner-ipv6-raw
- lag hash speculate-mpls inner-ipv6-tag

For more information, see "MPLS transit load balancing" under "LAG load sharing" in the *Extreme SLX-OS Layer 2 Switching Configuration Guide*.

Setting global MPLS policy parameters

The user can optionally set the following global MPLS policy parameters (they apply to all MPLS-enabled interfaces on the device):

- Retry time
- Retry limit
- Administrative group names
- Whether the device sends out IS-IS LSPs with TE extensions for its MPLS-enabled interfaces
- Configuring IP-over-MPLS TTL Propagation Control
- LSP Accounting

Setting the retry limit

When the ingress LER fails to connect to the egress LER in a signaled LSP, the ingress LER tries indefinitely to make the connection unless the user sets a limit for these connection attempts.

After this limit is exceeded, the ingress LER stops trying to connect to the egress LER over the primary path. When a secondary path is configured for the LSP, it is immediately activated after the primary path fails. After the secondary path is activated, the ingress LER continues to try to connect to the egress LER over the primary path either up to the configured retry limit or indefinitely when no retry limit is set. When a connection over the primary path can be established, the secondary path is deactivated, and traffic for the LSP is again sent over the primary path.

To set the number of collection attempts, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable policy parameter configuration.

```
device(config-router-mpls)# policy
```

4. Configure the retry limit.

```
device(config-mpls-policy)# retry-limit 20
```

   In this example, the retry limit is configured to 20.

In the following example, the retry limit is configured with a value of 20.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# retry-limit 20
```

Once the connection is established, the retry counter is reset to zero. In the example above, when an LSP needs to be established again, the ingress LER makes 20 attempts to establish a connection to the egress LER.

Setting the retry time

When a signaled LSP is enabled, the ingress LER attempts to connect to the egress LER over the primary path specified in the LSPs configuration. When the connection is not successful, by default the ingress LER waits 30 seconds before attempting the connection again. The user can configure the amount of time the ingress LER waits between connection attempts.

To specify, for example, a retry time of 45 seconds, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable to configure policy parameters.

```
device(config-router-mpls)# policy
```

4. Specify LSP connect retry time in seconds. In this example, it is set to 45 seconds.

```
device(config-mpls-policy)# retry-time 45
```

The following example shows how to set the retry time to 45 seconds.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# retry-time 45
```

Establishing administrative group names

Administrative groups, also known as resource classes or link colors, allows the user to assign MPLS-enabled interfaces to various classes.

When a device calculates the path for an LSP, it can take into account the administrative group to which a interface belongs; the user can specify which administrative groups the device can include or exclude when making its calculation.

Up to 32 administrative groups can be configured on the device. The user can see an administrative group either by its name or its number. Before the user can see an administrative group by its name, the user must specify a name for the group at the MPLS policy level and associate the name with that administrative group's number.

To establish three administrative group names, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable policy parameter configuration.

```
device(config-router-mpls)# policy
```

4. Enable administrative group name configuration.

```
device(config-router-mpls-policy)# admin-group gold 30
```

   In this example, the administrative group name of 'gold' is used with an administrative group number of '30'. The number has a range of 0-31.

5. Enable administrative group name configuration.

```
device(config-router-mpls-policy)# admin-group silver 20
```

   In this example, the administrative group name of 'silver' is used with an administrative group number of '20'. The number has a range of 0-31.

6. Enable administrative group name configuration.

```
device(config-router-mpls-policy)# admin-group bronze 10
```

   In this example, the administrative group name of 'bronze' is used with an administrative group number of '10'. The number has a range of 0-31.

In the following example, three administrative groups are configured.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# admin-group gold 30
device(config-router-mpls-policy)# admin-group silver 20
device(config-router-mpls-policy)# admin-group bronze 10
```

After the user associates an administrative group name with a number, the user can see it by name when assigning interfaces to the group or including or excluding the group from LSP calculations.

Enabling IS-IS LSPs with TE extensions for MPLS interfaces

When an MPLS-enabled device receives an IS-IS TE LSP, it stores the traffic engineering information in its Traffic Engineering Database (TED). The device uses information in the TED when performing calculations to determine a path for an LSP.

Information related to traffic engineering is carried in IS-IS traffic engineering LSPs. IS-IS TE LSPs have special extensions that contain information about an interface's administrative group memberships, IPv4 interface address, IPv4 neighbor address, maximum link bandwidth, reservable link bandwidth, unreserved bandwidth, and default traffic engineering metrics.

The user can configure the device to send out IS-IS TE LSPs to selected MPLS-enabled interfaces. To do this, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS policy configuration.

```
device(config-router-mpls)# policy
```

4. Enable IGP for advertising traffic engineering. In this example IS-IS traffic engineering for level 1 is selected.

```
device(config-router-mpls-policy)# traffic-engineering isis level-1
```

   The **level-1** option enables LSPs with TE extensions for the IS-IS level-1 domain. The **level-2** option enables LSPs with TE extensions for the IS-IS level-2 domain.

The following example configures the device to send out IS-IS-TE LSPs for all of its MPLS-enabled interface.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# traffic-engineering isis level-1
```

By default, the device does not send out IS-IS LSPs with TE extensions for its MPLS-enabled interfaces. Since information in the TED is used to make path selections using CSPF, and information in the TED comes from OSPF-TE LSAs or IS-IS LSPs with TE extensions, the user must enable the device to send out OSPF-TE LSAs or IS-IS LSPs with TE extensions when the user wants CSPF to perform constraint-based path selection.

Displaying information about IS-IS LSPs with TE extensions

To display information about IS-IS LSPs with TE extensions.

```
device# show isis database level2 detail
IS-IS Level-2 Link State Database
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
SLX-OS3.00-00           0x00000644   0x78e3        843               1/0/0
   Area Address:  49.0002
   NLPID:  CC(IP)
   Hostname:  SLX-OS3
   Auth: Len 17 MD5 Digest "c33db90a87b93c80111980dbd59a19ed"
   TE  Router ID: 15.15.15.15
   Metric:  10       IP-Extended 15.15.15.15/32      Up: 0 Subtlv: 0
   Metric:  10       IP-Extended 132.0.0.0/24        Up: 0 Subtlv: 0
   Metric:  10       IP-Extended 121.0.0.0/24        Up: 0 Subtlv: 0
   Metric:  10       IS-Extended PE4.06
     Admin Group: 0x00000000
     Interface IP Address: 121.0.0.2
     Link BW: 1000000 kbits/sec
     Reservable BW: 1000000 kbits/sec
     Unreserved BW:
       [0]    1000000 kbits/sec  [1]    1000000 kbits/sec
       [2]    1000000 kbits/sec  [3]    1000000 kbits/sec
       [4]    1000000 kbits/sec  [5]    1000000 kbits/sec
       [6]    1000000 kbits/sec  [7]    1000000 kbits/sec
   Metric:  10       IS-Extended SLX-OS4.00
     Admin Group: 0x00000000
     Interface IP Address: 132.0.0.2
     Neighbor IP Address: 132.0.0.1
     Link BW: 10000000 kbits/sec
     Reservable BW: 10000000 kbits/sec
     Unreserved BW:
       [0]   10000000 kbits/sec  [1]   10000000 kbits/sec
       [2]   10000000 kbits/sec  [3]   10000000 kbits/sec
       [4]   10000000 kbits/sec  [5]   10000000 kbits/sec
       [6]   10000000 kbits/sec  [7]   10000000 kbits/sec
```

Configuring CSPF interface constraint

Under the default condition, hops configured as interface addresses in an LSP path are resolved to the router ID.

An LSP can be configured that does not traverse a specified interface. The **cspf-interface-constraint** command forces the CSPF calculation to include any specified interface when creating an LSP.

To configure the device to always include a specified interface when forming an LSP, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS policy parameter configuration.

```
device(config-router-mpls)# policy
```

4. Configure interface IP address for CSPF computation.

```
device(config-router-mpls-policy)# cspf-interface-constraint
```

The following example shows configuration of the device to always include a specified interface when forming an LSP.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# cspf-interface-constraint
```

The CSPF interface constraint feature may be dynamically turned on or off. Turning the feature off or on has no effect on LSPs that have already been established (primary and secondary). For LSPs that are currently retried, changing the constraint setting changes the behavior on the next retry such as when an LSP whose path is configured to use that interface fails to come up due to an interface down condition.

> NOTE
> The CSPF interface Constraint feature has significance for the ingress node only, where the CSPF calculation takes place for an LSP or a detour segment.

Setting traffic engineering parameters for MPLS interfaces

When using constraints to determine a path for an LSP, the device takes into account information included in IS-IS LSPs with TE extensions. This information can be used to set up a path for a new LSP or to preempt an existing LSP so that an LSP with a higher priority can be established.

IS-IS LSPs with TE extensions include *Type/Length/Value triplets (TLVs)* containing the following information:

- IP address of the local interface
- IP address of the remote interface (must exist with point-to-point links)
- Traffic engineering metric for the link
- Maximum bandwidth on the interface
- Maximum reservable bandwidth on the interface
- Unreserved bandwidth on the interface
- Administrative groups to which the interface belongs

. When configured to do so with the **traffic-engineering isis** command, the device sends out IS-IS LSPs containing this TE information for each of its MPLS-enabled interfaces. Optionally, the user can specify the maximum amount of bandwidth that can be reserved on an interface. In addition, the user can assign interfaces to administrative groups.

Reserving bandwidth on an interface

IS-IS LSPs with TE extensions contain three TLVs related to bandwidth reservation:

- The maximum bandwidth TLV indicates the maximum outbound bandwidth that can be used on the interface. Maximum bandwidth is the operating speed of the port. When calculated for a LAG, the Maximum Bandwidth is the operating speed of the primary port multiplied by the number of active ports in the LAG. This reflects the actual physical bandwidth of the interface. This TLV is not configurable by the user.
- The maximum reservable bandwidth TLV indicates the maximum bandwidth that can be reserved on the interface. By default, the maximum reservable bandwidth is the same as the maximum bandwidth for the interface. The user can optionally change the reservable bandwidth to an amount greater or less than the maximum available bandwidth of the interface. When a maximum reservable bandwidth is configured on the primary port within a LAG, the value configured applies to the entire LAG regardless of any change to the number of active ports within the LAG. By default, the maximum reservable bandwidth for the LAG is the same as its maximum bandwidth.
- The unreserved bandwidth TLV indicates the amount of bandwidth not yet reserved on the interface. This TLV consists of eight octets, indicating the amount of unreserved bandwidth (in kilobits per second) at each of eight priority levels. The octets correspond to the bandwidth that can be reserved with a hold priority of 0 through 7, arranged in increasing order, with priority 0 occurring at the start of the TLV, and priority 7 at the end of the TLV. The value in each of the octets is less than or equal to the maximum reservable bandwidth. The unreserved bandwidth TLV itself is not user-configurable, although it is affected by modifications to the reservable bandwidth on an interface, as well as changes to LSPs.

Optionally, the user can change the amount of reservable bandwidth on an MPLS-enabled interface (that is, modify the value in the maximum reservable bandwidth TLV in IS-IS TE LSPs sent out for the interface). The maximum reservable bandwidth on an MPLS-enabled interface can be configured in either of two ways: as an absolute value, or as a percentage of the total interface bandwidth.

Reservable bandwidth configuration considerations

The **reservable-bandwidth** command is configurable on an MPLS-enabled interface at any time. The configuration of the command takes effect immediately upon preemption of the LSP.

When LSP preemption occurs, when the reservable bandwidth required for a specific LSP is not supported on the interface, then the LSP immediately goes down. When this occurs, an IGP advertisement of this configuration change is triggered and flooded throughout all ports on the network because the maximum reservable bandwidth configured on the interface is different from the value that was previously configured.

> NOTE
> When the maximum reservable bandwidth is configured as a percentage value for LAGs and VE interfaces, and ports go down, or new ports are added to the interface, the reservable bandwidth is recalculated as a percentage of the newly available bandwidth for that interface.

Configuring the maximum reservable bandwidth

To configure the maximum reservable bandwidth as an absolute value for MPLS LSPs on the interface, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces.

```
device(config-router-mpls)# mpls-interface ethernet 1/1
```

4. Enable LDP parameters

```
device(config-router-mpls-if-eth-1/1)# ldp-params
```

5. Configure the maximum reservable bandwidth in kbps.

```
device(config-router-mpls-if-eth-1/1-ldp-params)# reservable-bandwidth 10000
```

   In this example, the maximum reservable bandwidth is configured to 10000 kbps.

The following example shows the configuration of the maximum reservable bandwidth for MPLS LSPs with an absolute value of 10000 kbps.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 1/1
device(config-router-mpls-if-eth-1/1)# ldp-params
device(config-router-mpls-if-eth-1/1-ldp-params)# reservable-bandwidth 10000
```

Configuring the maximum reservable bandwidth as a percentage

To configure the maximum reservable bandwidth as a percentage of the total interface bandwidth for MPLS LSPs on the interface, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces.

```
device(config-router-mpls)# mpls-interface ethernet 1/1
```

4. Enable LDP configuration.

```
device(config-router-mpls-if-eth-1/1)# ldp-params
```

5. Configure the maximum reservable bandwidth with the percentage value of the interface bandwidth that can be used by MPLS LSPs, when requested.

```
device(config-router-mpls-if-eth-1/1-ldp-params)# reservable-bandwidth percentage 80
```

The percentage value of 100 specifies that the entire interface bandwidth can be used by MPLS LSPs when needed.

When maximum reservable bandwidth is changed from an absolute value to a percentage value, and vice versa, the following advisory message is displayed on the console to indicate this configuration change.

```
device(config-router-mpls-if-eth-1/1-ldp-params)# reservable-bandwidth percentage 40
Maximum reservable bandwidth is changed from 30 kbps to 40%
```

NOTE

When the maximum reservable bandwidth is configured as either an absolute value, or a percentage value, the value is recalculated and updated to the latest value.

To set the maximum reservable bandwidth back to the default value (the total physical bandwidth of the interface) when the absolute value or percentage value is used, enter the **no** form of the command as displayed in the following example.

```
device(config-router-mpls-if-eth-1/1-ldp-params)# no reservable-bandwidth percentage 80
```

By default, the reservable bandwidth is the same as the maximum available bandwidth on the interface. When the amount of reservable bandwidth is greater than the maximum available bandwidth, then the link can be oversubscribed. When the reservable bandwidth is less than the maximum available bandwidth, then LSPs cannot reserve all physical bandwidth on the interface. When the **reservable bandwidth** command is applied to the primary port within a LAG, the bandwidth configured for that port applies to the entire LAG, regardless of any change to the number of active ports within the LAG.

Changing the amount of reservable bandwidth on an interface causes the amount of unreserved bandwidth to be recalculated. In addition, it may cause an IS-IS-TE LSP to be issued, as well as possibly pre-empt existing LSPs when bandwidth reservations can no longer accommodate them.

The output from the **show running-config router mpls** command and the **show running-config router mpls mpls-interface ethernet** *slot/port* command displays the maximum reservable bandwidth configuration. Depending on the interface configuration, the show commands displays the maximum reservable bandwidth as an absolute value, or as a percentage value. When the **no** form of the **reservable-bandwidth** command is used, the default value of the interface bandwidth is also displayed in both show command outputs.

Configuration considerations

The **reservable-bandwidth** command is configurable on an MPLS-enabled interface at any time. The configuration of the command takes effect immediately upon preemption of the LSP. When LSP preemption occurs, when the reservable bandwidth required for a specific LSP is not supported on the interface, then the LSP immediately goes down. When this occurs, an IGP advertisement of this configuration change is triggered and flooded throughout all ports on the network because the maximum reservable bandwidth configured on the interface is different from the value that was previously configured.

NOTE

When the maximum reservable bandwidth is configured as a percentage value for LAGs and VE interfaces, and ports go down, or new ports are added to the interface, the reservable bandwidth is recalculated as a percentage of the newly available bandwidth for that interface.

To configure the maximum reservable bandwidth as an absolute value for MPLS LSPs on the interface, enter the following commands as displayed in the following example.

```
device# configure
device(config)# router mpls
```

```
device(config-router-mpls)# mpls-interface ethernet 1/1
device(config-router-mnpls-if-eth-1/1)# ldp-params
device(config-roouter-mpls-if-eth-1/1-ldp-params)# reservable-bandwidth 10000
```

To configure the maximum reservable bandwidth as a percentage of the total interface bandwidth for MPLS LSPs on the interface, enter the following commands as displayed in the following example.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 1/1
device(config-router-mpls-if-eth-1/1)# ldp-params
device(config-router-mpls-if-eth-1/1-ldp-params)# reservable-bandwidth percentage 80
```

When maximum reservable bandwidth is changed from an absolute value to a percentage value, and vice versa, the following advisory message is displayed on the console to indicate this configuration change.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 1/1
device(config-router-mpls-if-eth-1/1)# ldp-params
device(config-router-mpls-if-eth-1/1-ldp-params)# reservable-bandwidth percentage 40
Maximum reservable bandwidth is changed from 30 kbps to 40%
```

NOTE

When the maximum reservable bandwidth is configured as either an absolute value, or a percentage value, the value is recalculated and updated to the latest value.

To set the maximum reservable bandwidth back to the default value (the total physical bandwidth of the interface) when the absolute value or percentage value is used, enter the **no** form of the command as displayed in the following example.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 1/1
device(config-router-mpls-if-eth-1/1)# ldp-params
device(config-router-mpls-if-eth-1/1-ldp-params)# no reservable-bandwidth percentage 80
```

By default, the reservable bandwidth is the same as the maximum available bandwidth on the interface. When the amount of reservable bandwidth is greater than the maximum available bandwidth, then the link can be oversubscribed. When the reservable bandwidth is less than the maximum available bandwidth, then LSPs cannot reserve all physical bandwidth on the interface. When the **reservable-bandwidth** command is applied to the primary port within a LAG, the bandwidth configured for that port applies to the entire LAG, regardless of any change to the number of active ports within the LAG.

Changing the amount of reservable bandwidth on an interface causes the amount of unreserved bandwidth to be recalculated. In addition, it may cause an IS-IS TE LSP to be issued, as well as possibly pre-empt existing LSPs when bandwidth reservations can no longer accommodate them.

The output from the **show running-config router mpls** command and the **show router mpls interface** [**ethernet** *slot/port*] command displays the maximum reservable bandwidth configuration. Depending on the interface configuration, the show commands displays the maximum reservable bandwidth as an absolute value, or as a percentage value. When the **no** form of the **reservable-bandwidth** command is used, the default value of the interface bandwidth is also displayed in both show command outputs.

Adding interfaces to administrative groups

Administrative groups, also known as resource classes or link colors, allows the user to assign MPLS-enabled interfaces to various classes.

The user can place individual interfaces into administrative groups. For example, the user can define a group called "gold" and assign high-bandwidth interfaces to it. When a device calculates the path for an LSP, it can take into account the administrative group to which a interface belongs. The user can configure up to 32 administrative groups. By default, an interface does not belong to any administrative groups.

Administrative groups are in the range 0 - 31. The user can see an administrative group either by name or number. To see an administrative group by name, first create a name for the group and associate the name with an administrative group number.

To assign MPLS-enabled interface e 3/1 to an administrative group called "gold", complete the following steps.

1. Configure the device.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable MPLS-capable interfaces.

   ```
   device(config-router-mpls)# mpls-interface ethernet 3/1
   ```

   In this example, ethernet interface 3/1 is selected.

4. Enable LDP parameters.

   ```
   device(config-router-mpls-if-eth-3/1)# ldp-params
   ```

5. Select administrative group.

   ```
   device(config-router-mpls-if-eth-3/1-ldp-params)# admin-group gold
   ```

   In this example, the administrative group 'gold' is selected.

In the following example, a MPLS-enabled interface (3/1) is added to the 'gold' administrative group.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 3/1
device(config-router-mpls-if-eth-3/1)# ldp-params
device(config-router-mpls-if-eth-3/1-ldp-params)# admin-group gold
```

A MPLS-enabled interface can belong to any number of administrative groups. For example, to assign an interface to group "gold" and group 31, enter commands such as the following.

```
device(config-router-mpls)# mpls-interface ethernet 3/1
device(config-router-mpls--if-eth-3/1)# ldp-params
device(config-router-mpls--if-eth-3/1-ldp-params)# admin-group gold 31
```

After the user adds interfaces to administrative groups, the user can specify which groups can be included or excluded from LSP calculations.

The MPLS process restart

The MPLS process restart capability is a fault containment mechanism which ensures that process-level failures do not cause system-level failures. To achieve process restart capability, each process must run in its protected memory space independent of kernel.

When a non-restartable process crashes, the MPLS process restart is to perform a failover to STANDBY in dual-MM chassis or perform a cold reboot in a Single-MM chassis.

MPLS has a COLD restartable process, meaning when there is a fault inside the MPLS process, and it crashes the system does not undergo a failover. Instead, the MPLS process is restarted on the same ACTIVE MM. All the other modules and processes that interact with MPLS are made aware of the MPLS restart, and they adjust accordingly. All the MPLS-based services, like IPoMPLS, VLL, VPLS are disrupted for the duration of MPLS process restart. Once the MPLS process is restarted, the control protocols (LDP and RSVP) re-signal the tunnels and cross-connects and subsequently, all the dependent MPLS applications resume service.

The MPLS cold process restart user-observable behavior

- All other non-MPLS services continue to function normally with any disruption due to MPLS process termination.
- Traffic loss on applications using MPLS services are down until the MPLS process restarts and re-signaling are complete. VLL/VPLS and BGP IPoMPLS services are affected during MPLS process restart.
- Any in progress (unacknowledged) MPLS configuration sessions will not be honored, and the user must reconfigure after the MPLS process comes back up. Attempts to configure or retrieve the operational state returns with an error.
- Once MPLS process restarts, it comes up with the previous running configuration and proceeds to establish all sessions and tunnels. The dependent MPLS services are restored.
- Existing configuration CLI sessions on MPLS process restart. Example: User was in the middle of configuring an LSP and MPLS process restarts, once MPLS is UP user can continue.
- Existing display CLI sessions with respect to MPLS are continued.
- No new MPLS configuration or display of operational information on MPLS is allowed during the time MPLS process is restarting.
- Graceful restart of LDP has no effect as the MPLS entries are cleaned up from the forwarding layer.
- The MPLS process restart failure results in failover to STANDBY with a COLD boot.

To following command disables the MPLS process restart.

```
device# configure
device(config)# ha
device(config-ha)# process-restart disable mpls
```

Traffic engineering database

An LSR TED stores topology information about the MPLS domain. This topology information comes from the IS-IS LSPs with TE extensions that are flooded throughout the IS-IS domain. When an LSR receives IS-IS LSPs with TE extensions from neighboring LSRs, it places the traffic engineering information into its TED.

LSP attributes and requirements used for traffic engineering

In addition to the topology information in the TED, the device considers attributes and requirements specified in configuration statements for the LSP. The following user-specified parameters are considered when the device calculates a traffic-engineered path for a signaled LSP:

- Destination address of the egress LER
- Explicit path to be used by the LSP
- Bandwidth required by the LSP
- Setup priority for the LSP
- Metric for the LSP
- Whether the LSP includes or excludes links belonging to specified administrative groups

Refer to Configuring signaled LSP parameters on page 134 for more information on how to set these parameters.

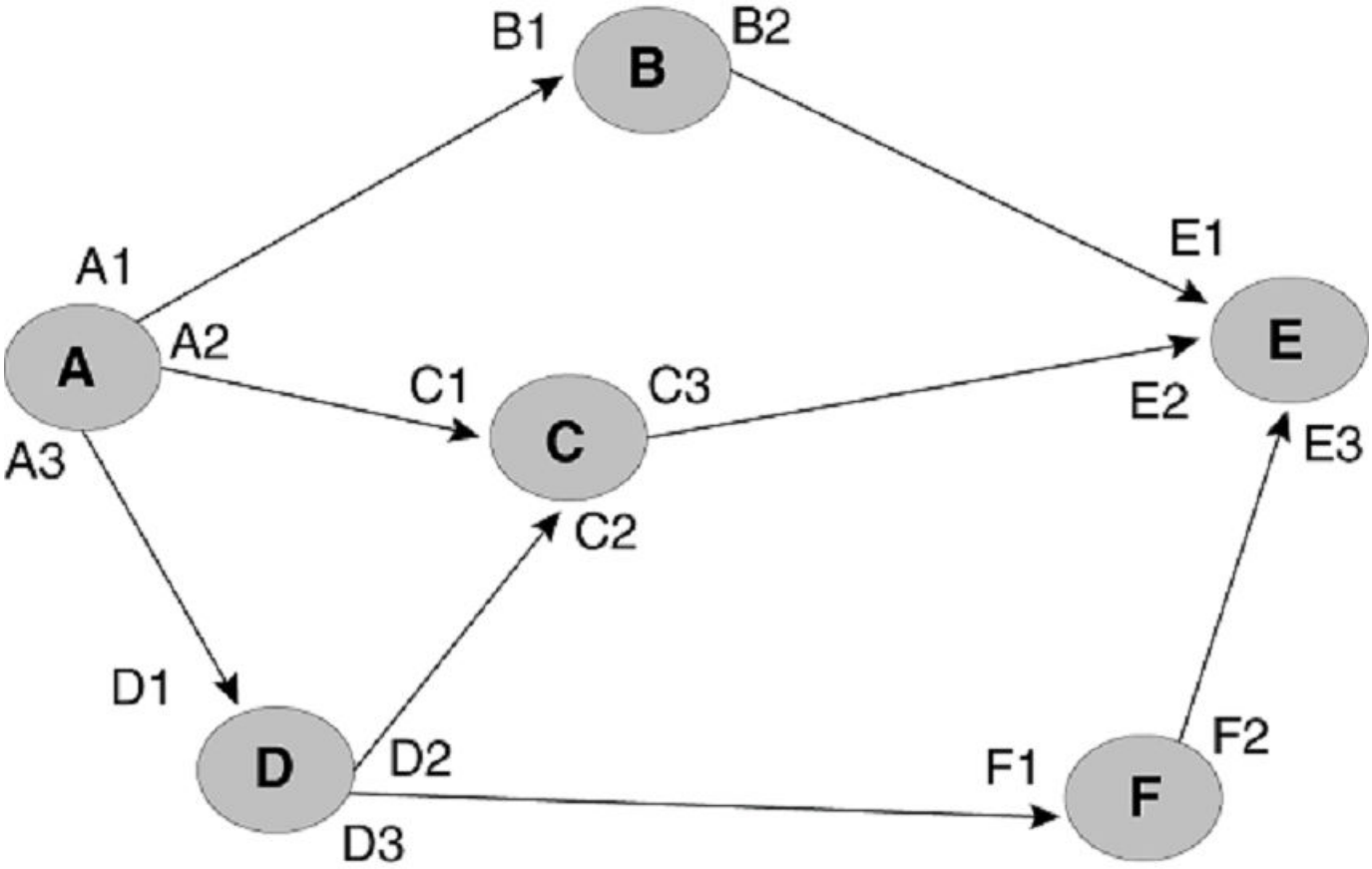
Calculating a path based on an interface address

Under normal conditions, router IDs are used to configure hops within an MPLS path. In situations where the user wants to exercise more control over the path, the user can specify actual interface addresses in the MPLS path to make sure that the path traverses the interface specified. In previous versions, the CSPF calculation would always resolve a specified interface address to the router ID. Consequently, although a particular interface on a router is specified, the CSPF calculation can end up connecting the path through a different interface on the router where the interface has been specified.

In the network described in the diagram below, the source node is "A" and the destination node is "E". In this configuration, incoming and outgoing interfaces are defined in the figure by their relationship to where the arrowhead on the connecting line points. The arrowheads point to the incoming interface from the outgoing interface. For instance "A1", "A2" and "A3" are the outgoing interfaces of node A and "C1" and "C2" are the incoming interfaces of node C. The following example describes how the router might calculate a path between "A" and "B" under the default operating condition.

In this example, an MPLS path has been configured with a source "A" and a destination "E1". Under default operation, the interface "E1" destination is resolved to the routerID for "E". This means that the path can be calculated to arrive at the "E" node on any of the following interfaces: "E1, "E2" or "E3". While a path that travels from node "A" to node "B" to node "E" is the only path that actually satisfies the intent of the configuration, any of the following paths could be created by CSPF under the default operation condition: "A" to "C" to "E", "A" to "D" to "C" to "E" or "A" to "D" to "F" to "E".

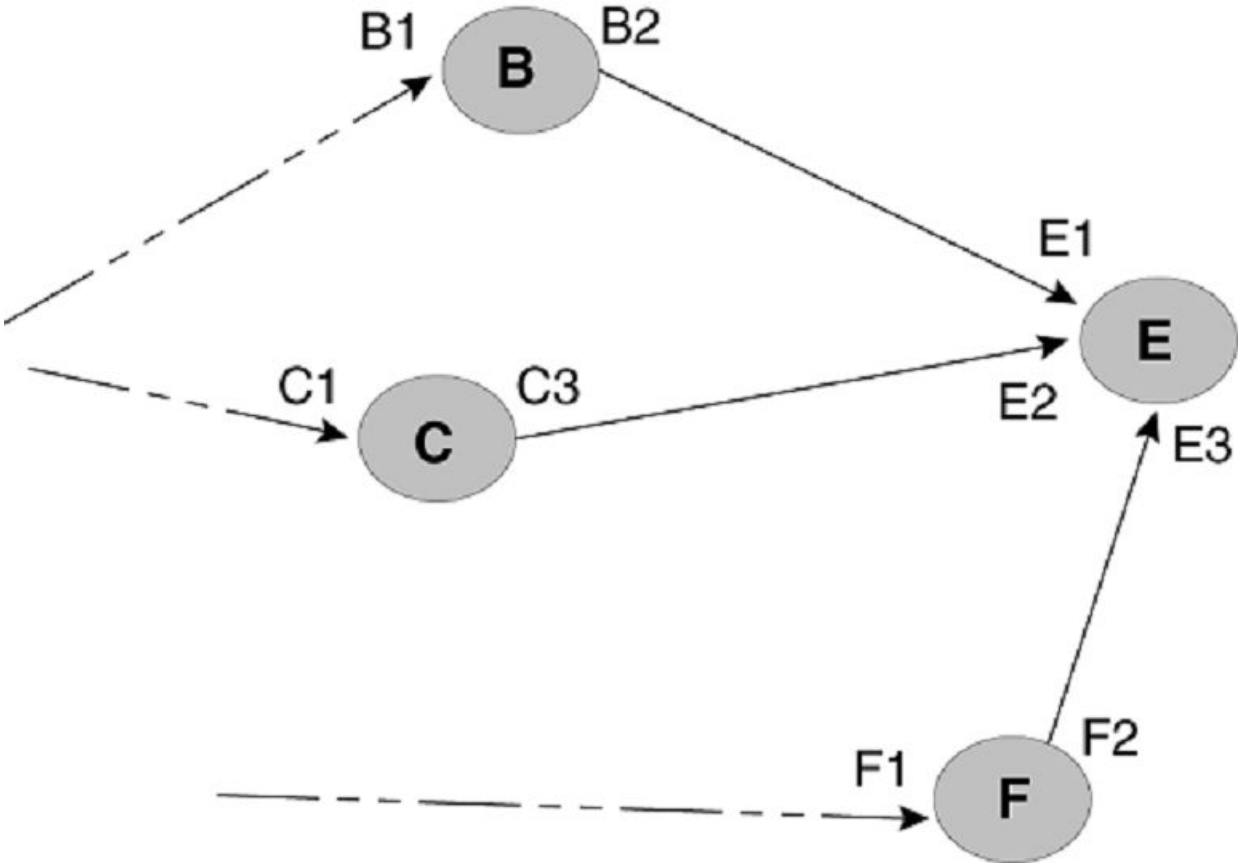
FIGURE 6 Calculating a path based on an interface



The global **cspf-interface-constraint** command directs the router to include the interface address as a constraint when it determines the shortest path. When invoked, this command ensures that a specified interface must be included in an LSP. This constraint can be turned on and off dynamically and does not affect established primary or secondary LSPs. CSPF interface constraint is significant for the ingress node only, where CSPF calculation takes place for an LSP.

When configuring CSPF interface constraint, the user must be aware that the imposition of this additional constraint can increase the possibility of no path being found where otherwise there could be a path. One case where this can occur is where the path required to conform to the interface constraint fails the configured bandwidth constraint. Additionally, no path may be found where a configured path contains an inherently contradictory condition. For example, when a path is configured "B1 (strict) to E2 (loose) as shown in the diagram below, no path is found. This is because CSPF always appends B1 into the final CSPF path. This has the effect of making "B" the source node of the next hop and therefore excludes "E1 as a traversed interface in subsequent paths to the destination node "E". Consequently, in this example the LSP is down. However, when the **cspf-interface-constraint** command is not active, a CSPF path is found and the LSP goes up.

FIGURE 7 Example of where no path is found



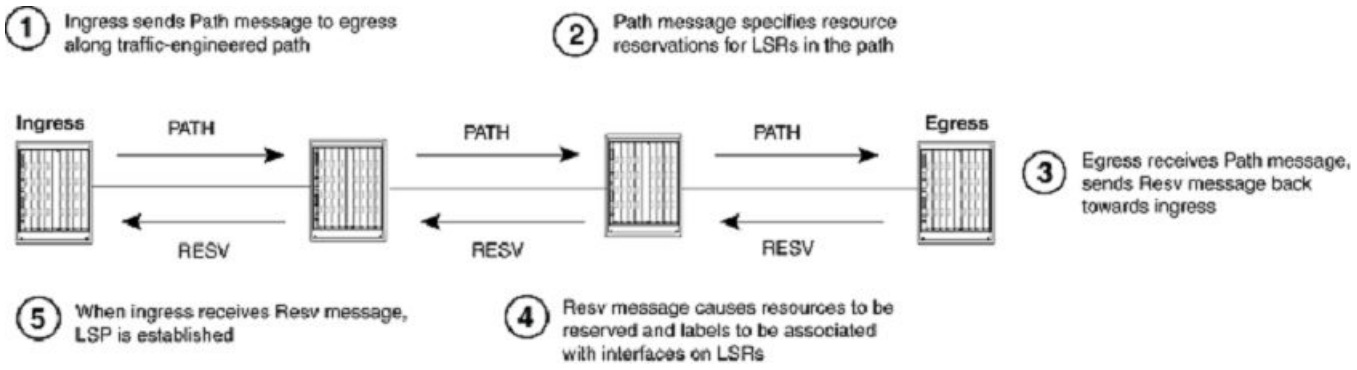
How RSVP establishes a signaled LSP

The traffic-engineered path calculated by CSPF consists of a sequential list of physical interface addresses, corresponding to a path from the ingress LER to the egress LER. Using this traffic-engineered path, RSVP establishes the forwarding state and resource reservations on each LSR in the path.

Special extensions for traffic engineering are defined for RSVP. These extensions include the EXPLICIT_ROUTE, LABEL_REQUEST, LABEL, and RECORD_ROUTE objects in addition to the *Fixed Filter (FF)* reservation style. These extensions are described in *RFC 3209* .

The following diagram illustrates how RSVP establishes a signaled LSP.

FIGURE 8 How RSVP establishes a signaled LSP



RSVP signaling for LSPs works as described below.

1. The ingress LER sends an RSVP Path message towards the egress LER.

   The Path message contains the traffic engineered path calculated by the CSPF process, specified as an *EXPLICIT_ROUTE object (ERO)* . The Path message travels to the egress LER along the route specified in the ERO.

   The Path message also describes the traffic for which resources are being requested and specifies the bandwidth that needs to be reserved to accommodate this traffic. In addition, the Path message includes a LABEL_REQUEST object, which requests that labels be allocated on LSRs and tells the egress LER to place a LABEL object in the Resv message that it sends back to the ingress LER.

   Before sending the Path message, the ingress LSR performs admission control on the outbound interface, ensuring that enough bandwidth can be reserved on the interface to meet the LSPs requirements. Admission control examines the LSPs configured setup priority and mean-rate settings. For the LSP to pass admission control, the outbound interface must have reservable bandwidth at the LSPs setup priority level that is greater than the amount of bandwidth specified by the LSPs mean-rate setting. Refer to Admission control, bandwidth allocation, and LSP preemption on page 48, for more information and examples of this process.
2. The Path message requests resource reservations on the LSRs along the path specified in the ERO.

   When the LSP passes admission control, the ingress LER sends a Path message to the address at the top of the ERO list. This is the address of a physical interface on the next LSR in the path. As the ingress LER did, this LSR performs admission control to make sure the outbound interface has enough reservable bandwidth to accommodate the LSP.

   When the LSP passes admission control, the LSR then removes its address from the top of the ERO list and sends the Path message to the address now at the top of the ERO list. This process repeats until the Path message reaches the last node in the ERO list, which is the egress LER.
3. The egress LER receives the Path message and sends a Resv message towards the ingress LER.

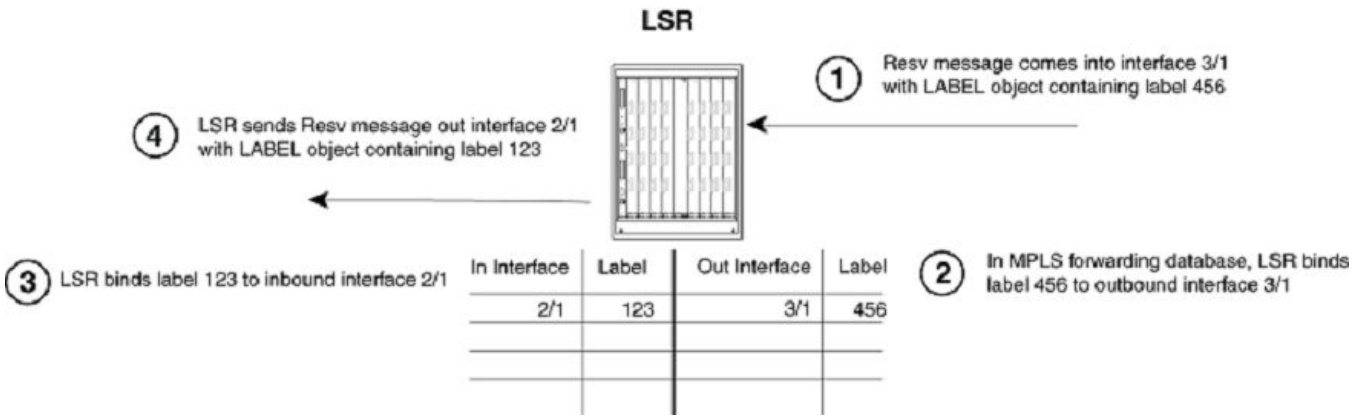
   Resv messages flow upstream from the receiver of the Path message to the sender (that is, from the egress LER to the ingress LER), taking the exact reverse of the path specified in the ERO. In response to the LABEL_REQUEST object in the Path message, the Resv message from the egress LER includes a LABEL object. The LABEL object is used to associate labels with interfaces on the LSRs that make up the LSP.

4. As the Resv messages travel upstream, resources are reserved on each LSR.

   When an LSR receives a Resv message, it again performs admission control on the interface where the Resv message was received (that is, the interface that is the outbound interface for packets traveling through the LSP). When the LSP still passes admission control, bandwidth is allocated to the LSP. The LSR allocates the amount of bandwidth specified by the LSPs mean-rate setting, using bandwidth available to its hold priority level. This may cause lower priority LSPs active on the device to be preempted.

   Once bandwidth has been allocated to the LSP, the LABEL object in the Resv message is used to associate labels with interfaces in the LSRs MPLS forwarding table. Figure 9 shows an example of how this works.

   FIGURE 9 How the RSVP LABEL object associates a label with an interface in the MPLS forwarding table



   In the example above, the LSR receives a Resv message on interface 3/1 from the downstream LSR in the ERO. The Resv message has a LABEL object containing label 456. After performing admission control and bandwidth allocation, the LSR adds an entry to its MPLS forwarding table for this LSP, associating label 456 with outbound interface 3/1.

   The LSR then takes a label from its range of available labels (for example, 123) and places it in the LABEL object in the Resv message that it sends to the upstream LSR. In this example, the LSR sends the Resv message out interface 2/1 to the upstream LSR in the ERO. In its MPLS forwarding table for this LSP, the LSR associates label 123 with inbound interface 2/1.

   This process repeats at each LSR until the Resv message reaches the ingress LER.

   NOTE

   To enable penultimate hop popping for the LSP, the LABEL object sent by the egress LER to the penultimate LSR contains a value of three (3) (Implicit Null Label). This is an IETF-reserved label value that indicates to the penultimate LSR that it must pop the label of MPLS-encoded packets that belong to this LSP.

5. Once the Resv message reaches the ingress LER, and the process described in Step 4 takes place, the LSP is activated. At this point each LSR in the LSP has reserved resources, allocated labels, and associated labels with interfaces. The LSP is activated, and the ingress LER can assign packets to the LSP.

Refresh messages

Once a signaled LSP is enabled at the ingress LER, the router persistently attempts to establish the LSP through periodic retries until the LSP is successfully established. To maintain the forwarding states and resource reservations on the routers in an LSP, Path and Resv messages are exchanged between neighboring LSRs at regular intervals. When these refresh messages are not received on the routers in the LSP, the RSVP forwarding states and resource reservations are removed. The user can control how often the Path and Resv messages are sent, as well as how long the device waits before removing forwarding states and resource reservations. The user can also use reliable messaging and refresh reduction to reduce RSVP message bandwidth and improve the dependability of RSVP paths and reservations states.

Admission control, bandwidth allocation, and LSP preemption

When a Resv message is received on an LSR, admission control determines whether the LSP can be established, based on its configured priority. When an LSP passes admission control, bandwidth is allocated to the new LSP, possibly preempting existing LSPs that have lower priority.

An LSPs priority consists of a setup priority and a hold priority. The setup priority is the priority for taking resources; the hold priority is the priority for holding resources. An LSPs setup priority is considered during admission control, and its hold priority is considered when bandwidth is allocated to the LSP. The setup and hold priorities are expressed as numbers between zero (0) (highest priority level) and seven (7)(lowest priority level). An LSPs setup priority must be lower than or equal to its hold priority. The user can configure either of these values for an LSP; by default, an LSPs setup priority is seven and its hold priority is zero.

On an MPLS-enabled interface, a certain amount of bandwidth is allocated for usage by LSPs; this amount can be either the maximum available bandwidth on the interface (the default) or a user-specified portion. The amount of bandwidth an individual LSP can reserve from this pool of allocated bandwidth depends on two user-configured attributes of the LSP: the LSPs priority and the LSPs mean-rate (the average rate of packets that can go through the LSP). The following conditions also apply:

- For an LSP to pass admission control, the bandwidth available to its setup priority level must be greater than the value specified by its mean-rate.
- When an LSP passes admission control, the bandwidth specified by its mean-rate is allocated to the LSP, using bandwidth available to its hold priority level.
- For the allocation of bandwidth to the new LSP, the system might preempt existing, lower-priority LSPs.

When setting up an LSP, the device actually performs admission control twice: when the Path message is received and when the Resv message is received. when the LSP passes admission control after the Resv message is received, bandwidth allocation and LSP preemption take place.

The sections that follow include examples of how admission control, bandwidth allocation, and preemption work.

Admission control

Admission control examines the LSPs setup priority and mean-rate settings to determine whether the LSP can be activated. To pass admission control, the reservable bandwidth available at the LSPs setup priority level must be greater than the value specified by its mean-rate.

For example, when the maximum reservable bandwidth on an interface is 10,000 Kbps and no LSPs are currently active, the amount of reservable bandwidth on the interface for each priority level would be as follows:

TABLE 3 LSP setup priority and mean-rate settings

Priority	Unreserved Bandwidth
0	10,000
1	10,000

TABLE 3 LSP setup priority and mean-rate settings (continued)

2	10,000
3	10,000
4	10,000
5	10,000
6	10,000
7	10,000
Active LSPs : None	

The LSR receives a Resv message for an LSP that has a configured setup priority of six and a hold priority of three. The mean-rate specified for this LSP is 1,000 Kbps. For priority level 6, up to 10,000 Kbps can be reserved. Because the configured mean-rate for this LSP is only 1,000 Kbps, the new LSP passes admission control.

Bandwidth allocation

Once the LSP passes admission control, bandwidth is allocated to it. The bandwidth allocation procedure examines the LSPs hold priority and mean-rate settings. The amount of bandwidth specified by the mean-rate is allocated to the LSP, using reservable bandwidth available at the LSPs hold priority level.

In this example, the LSPs hold priority is three and mean-rate is 1,000 Kbps. On this interface, for priority level three, up to 10,000 Kbps can be reserved. The amount of bandwidth specified by the mean-rate (1,000 Kbps) is allocated to the LSP.

After bandwidth is allocated to this LSP, the amount of unreserved bandwidth on the interface is reduced accordingly. In the example, the reservable bandwidth array for the interface now looks like this:

TABLE 4 Bandwidth allocations

Priority	Unreserved Bandwidth
0	10,000
1	10,000
2	10,000
3	9,000
4	9,000
5	9,000
6	9,000
7	9,000
Active : LSP with setup 6, hold 3, mean-rate 1,000	

Given the bandwidth allocation above, when an LSP is established with a setup priority of three and a mean-rate of 9,500 Kbps, it would not pass admission control because only 9,000 Kbps is available at priority 3.

LSP preemption

When there is not enough unallocated bandwidth on an interface to fulfill the requirements of a new LSP that has passed admission control, existing LSPs that have a lower priority may be preempted. When preemption occurs, bandwidth allocated to lower-priority LSPs is reallocated to the higher-priority LSP. LSP preemption depends on the bandwidth requirements and priority of the new LSP, compared to the bandwidth allocation and priority of already existing LSPs.

When LSP preemption is necessary, the device uses the following rules:

NOTE
LSP preemption rules have changed to improve the scalability and performance for *Fast Reroute (FRR)* enabled LSPs. See bullets three and four below for changes to LSP preemption for FRR enabled LSPs.

- Preempt existing LSPs that have lower priority than the new LSP
- When several existing LSPs have lower priority than the new LSP, preempt the LSP that has the lowest priority
- When two LSPs have equal priority and one LSP must be preempted, preempt the LSP which is currently FRR enabled irrespective of its bandwidth requirement
- Preempt as many FRR enabled LSPs as necessary before preempting unprotected LSPs of the same priority. For example, when both FRR enabled LSPs and non-FRR enabled LSPs are configured, the system attempts its best to preempt FRR enabled LSPs first before preempting non-FRR enabled LSPs until the bandwidth requirement is met for a new high priority LSP

In the example above, bandwidth has been allocated to an LSP that has a hold priority of three and a mean-rate of 1,000 Kbps. When a new LSP with a setup priority of two, hold priority of one, and mean-rate of 10,000 Kbps is established, admission control, bandwidth allocation, and LSP preemption work as described below.

1. **Admission control:** On the interface, there is 10,000 Kbps available to priority two. The mean-rate for the new LSP is 10,000, so the LSP passes admission control; bandwidth can be allocated to it.
2. **Bandwidth allocation:** The hold priority for the new LSP is one. On the interface, 10,000 Kbps is available to priority one. This entire amount is allocated to the LSP.
3. **LSP preemption:** The first LSP had been using 1,000 Kbps of this amount, but its hold priority is only three. Consequently, the first LSP is preempted, and its bandwidth allocation removed in order to make room for the new LSP.

   Once this happens, the reservable bandwidth array for the interface looks like this:

   TABLE 5 LSP preemption bandwidth allocations

Priority	Unreserved Bandwidth
0	10,000
1	0
2	0
3	0
4	0
5	0
6	0
7	0
Active : LSP with setup 2, hold 1, mean-rate 10,000 **Preempted:** LSP with setup 6, hold 3, mean-rate 1,000	

   On this interface, the only LSP that could preempt the active LSP would be have a setup and hold priority of zero.

   When multiple LSPs are candidates for preemption, the device normally preempts the LSP with the lowest priority. However, when preempting a higher priority LSP with a high bandwidth requirement would allow lower priority LSPs with lower bandwidth requirements to avoid preemption, the higher-priority LSP is preempted.

   For example, consider an interface with 10,000 Kbps of reservable bandwidth, allocated to two active LSPs: one with a setup priority of three, hold priority of two, and mean-rate of 5,000 Kbps; and another with a setup priority of four, hold priority of three, and mean-rate of 2,500 Kbps. When an LSP with a setup priority of one, hold priority of zero, and mean-rate of 7,500 Kbps is established, the following take place.

MPLS traffic engineering flooding reduction

Traffic engineering advertisements are triggered when a threshold value is reached or crossed. For all other bandwidth changes, a periodic flooding timer or *Connection Admission Check (CAC)* failure triggers the TE advertisements. When no thresholds are crossed, changes are flooded periodically unless periodic flooding was disabled. Configurations can be executed as a global configuration or interface specific configuration.

Interface specific configurations supersedes global configuration and default values. Global configuration supersedes default values. When there is no interface specific configuration and global configuration, then the default values are used.

MPLS traffic engineering flooding reduction global configuration

Reserved bandwidth threshold configuration can be executed globally and is applied to all MPLS interfaces. Global configurations are done at the policy mode under router mpls.

To set RSVP-TE flooding thresholds at the global configuration level, complete the following steps.

1. Enable the device and configure the terminal to the global configuration mode.

```
device>enable
device# configure terminal
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the policy parameters.

```
device(config-mpls)# policy
```

4. Enable RSVP triggered traffic engineering LSA flooding reduction. Use the **up** option. This triggers the bandwidth percentage to go up when the bandwidth is increased.

```
device(config-mpls-policy)# rsvp-flooding-threshold up 10 20 30 40 50 55 60 65 70
85 90 92 93 94 95 96 97 98 99 100
```

   The default for the **up** option is 15, 30, 45, 60, 75, 80, 85, 90, 95, 96, 97, 98, 99, and 100.

5. Enable RSVP triggered traffic engineering LSA flooding reduction. Use the **down** option. This triggers the bandwidth percentage to go down when the bandwidth is decreased.

```
device(config-mpls-policy)# rsvp-flooding-threshold down 99 98 97 96 95 94 93 92
91 90 85 80 75 70 65 60 55 50 40 30 20 10
```

   The default for the **down** option is 99, 98, 97, 96, 95, 90, 85, 80, 75, 60, 45, 30, and 15.

The **rsvp-flooding-threshold** command can be executed multiple times in the policy mode; the threshold values are added to the existing set of global threshold values. The previously configured values are not overwritten.

The following example shows the threshold configurations be executed globally.

```
device>enable
device#configure terminal
device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# rsvp-flooding-threshold up 10 20 30 40 50 55 60 65 70
85 90 92 93 94 95 96 97 98 99 100
device(config-mpls-policy)# rsvp-flooding-threshold down 99 98 97 96 95 94 93 92
91 90 85 80 75 70 65 60 55 50 40 30 20 10
```

In the following example, the UP threshold contains 10, 50, 55, 95, 96, 97, 98, 99, and 100. The DOWN threshold contains 50, 40, 30, 20, and 10.

```
device>enable
device# configure terminal
device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# rsvp-flooding-threshold up 10 50 55 95 96
device(config-mpls-policy)# rsvp-flooding-threshold up 97 98 99 100
device(config-mpls-policy)# rsvp-flooding-threshold down 50 40 30
device(config-mpls-policy)# rsvp-flooding-threshold down 20 10
device(config-mpls-policy)#
```

MPLS traffic engineering flooding reduction interface specific configuration

The **rsvp-flooding-threshold** command can be executed multiple times for the same interface. The threshold values are added to the existing set of values for the interface. Previously configured values are not overwritten. The interface specific configuration overrides the global configuration. Using the **no** form of the command removes the sub-set of the configured threshold values.

Complete the following steps using the **rsvp-flooding-threshold** command at the MPLS interface level to set the reserved bandwidth threshold.

1. Enable device and configure the terminal to the global configuration mode.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable MPLS-capable interfaces.

   ```
   device(config-router-mpls)# mpls-interface ethernet 1/1
   ```

   In this example, ethernet interface 1/1 is enabled.

4. Enable LDP parameters.

   ```
   device(config-mpls-if-eth-1/1)# ldp-params
   ```

5. Enable RSVP triggered traffic engineering LSA flooding reduction. Use the **up** option. This triggers the bandwidth percentage to go up when the bandwidth is increased.

   ```
   device(config-mpls-if-eth-1/1-ldp-params)# rsvp-flooding-threshold up 10 20 30 40 50 55 60
   65 70 85 90 92 93 94 95 96 97 98 99 100
   ```

6. Enable RSVP triggered traffic engineering LSA flooding reduction. Use the **down** option. This triggers the bandwidth percentage to go down when the bandwidth is decreased.

```
device(config-mpls-if-eth-1/1-ldp-params)# rsvp-flooding-threshold down 99 98 97 96 95 94
93 92 91 90 85 80 75 70 65 60 55 50 40 30 20 10
```

the following example shows how to set the reserved bandwidth threshold at the interface level.

```
device# configure
device(config)# router mpls
device(config-mpls)# mpls-interface ethernet 1/1
device(config-mpls-if-eth-1/1)# ldp-params
device(config-mpls-if-eth-1/1-ldp-params)# rsvp-flooding-threshold up 10 20 30 40 50 55 60
65 70 85 90 92 93 94 95 96 97 98 99 100
device(config-mpls-if-eth-1/1-ldp-params)# rsvp-flooding-threshold down 99 98 97 96 95 94
93 92 91 90 85 80 75 70 65 60 55 50 40 30 20 10
device(config-mpls-if-eth-1/1-ldp-params)#
```

In the following example, the UP thresholds contain 10, 50, 55, 95, 96, 97, 98, and 100. The DOWN thresholds contain 50, 40, 30, 20, and 10.

```
device# configure
device(config)# router mpls
device(config-mpls)# mpls-interface ethernet 1/1
device(config-mpls-if-eth-1/1-ldp-params)# ldp-params
device(config-mpls-if-eth-1/1-ldp-params)# rsvp-flooding-threshold up 10 50 55 95
device(config-mpls-if-eth-1/1-ldp-params)# rsvp-flooding-threshold up 96 97 98 99 100
device(config-mpls-if-eth-1/1-ldp-params)# rsvp-flooding-threshold down 50 40
device(config-mpls-if-eth-1/1-ldp-params)# rsvp-flooding-threshold down 30 20 10
```

MPLS traffic engineering flooding reduction configuring the periodic flooding timer

All MPLS interfaces are checked every three minutes by default. TE advertisements are triggered when there is a difference in the available bandwidth and advertised available bandwidth.

Use the **rsvp-periodic-flooding-timer** command to set the interval for periodic flooding. The interval is set in seconds. To set the interval as 240 which triggers periodic flooding every four minutes, complete the following steps.

1. Enable the device and configure the terminal to the global configuration mode.

```
device>enable
device# configure terminal
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure policy parameters.

```
device(config-mpls)# policy
```

4. Set the interval for RSVP TE periodic flooding.

```
device(config-mpls-policy)# rsvp-periodic-flooding-timer 240
```

   In this example, the timer is set to 240 seconds (four minutes).

The following example shows how to set the periodic flooding timer. In this example, the time is set for 240 seconds (four minutes).

```
device>enable
device# configure terminal
device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# rsvp-periodic-flooding-timer 240
device(config-mpls-policy)# no rsvp-periodic-flooding-timer
```

RSVP soft preemption

RSVP soft preemption implements a suite of protocol modifications extending the concept of preemption with the goal of reducing or eliminating traffic disruption of TE LSPs. It is achieved by using additional signaling and maintenance mechanisms to alert the ingress LER of the preemption that is pending and allows for temporary control-plane under-provisioning while the preempted tunnel is rerouted in a non-disruptive fashion (make before-break) by the ingress LER. During the period that the tunnel is being rerouted, link capacity is under-provisioned on the midpoint where preemption was initiated and potentially one or more links upstream along the path where other soft preemptions may have occurred. Soft preemption is a property of the LSP and is disabled by default.

The default preemption in an MPLS-TE network is hard preemption. This is helpful in cases where actual resource contention happens in the network. Soft Preemption provides flexibility for operators to select the type of preemption based on network conditions.

MPLS soft preemption is useful for network maintenance. For example, all LSPs can be moved away from a particular interface, and then the interface can be taken down for maintenance without interrupting traffic. MPLS soft preemption is also useful in dynamic networks where preemption often occurs.

Only adaptive and non-FRR LSPs could be enabled for soft preemption. LSPs which are adaptive and without FRR configuration have the facility to enable or disable the soft preemption feature without disabling the LSP. When the soft preemption configuration is changed, RSVP is notified for this change and a new Path message is triggered with the soft preemption desired flag bit (0x40) set in session attribute for signaling.

Configuring RSVP soft preemption

Soft preemption capability on unprotected adaptive LSPs (which is disabled by default) can be configured irrespective of its state (enable or disable).

Non-adaptive and/or FRR enabled LSPs cannot be configured with soft preemption capability. In this scenario, the LSP must be disabled first to configure soft preemption based on the policies, other changes also may be required, such as removing FRR.

All secondary paths configured on the LSP would be allowed to have soft preemption configured independently.

The following steps must be followed to configure soft preemption.

1. Configure the device.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Define the path.

   ```
   device(config-router-mpls)# path sec
   ```

   This defines the path with the name of "sec".

4. Configure the LSP

```
device(config-router-mpls-path-sec)# lsp test
```

   This defines the LSP with a name of "test".

5. Set the egress router of the LSP.

```
device(config-router-mpls-lsp-test)# to 10.1.1.100
```

   This designates the egress router IP address of 10.1.1.100.

6. Configures the LSP as adaptive.

```
device(config-rouoter-mpls-lsp-test)# adaptive
```

   This enables the LSP to be dynamically modified.

7. Set the LSP to have preemption capability.

```
device(config-router-mpls-lsp-test)# soft-preemption
```

   This sets the primary path.

8. Configure the secondary path.

```
device(config-router-mpls-lsp-test)# secondary-path sec
```

9. When FRR is configured, remove the FRR configuration.

```
device(config-router-mpls-lsp-test)# traffic-eng mean-rate 100
```

10. Configure the secondary path to be adaptive.

```
device(config-router-mpls-lsp-test-secpath-sec)# adaptive
```

11. For each secondary path, where soft preemption is intended to be configured, mark them adaptive, when already not adaptive, configure SOFT preemption.

```
device(config-router-mpls-lsp-test-secpath-sec)# soft-preemption
```

    This sets the secondary path.

12. Enable the LSP.

```
device(config-router-mpls-lsp-test-secpath-sec)# enable
Connecting signaled LSP test
```

The **soft-preemption** command enables soft preemption functionality. This command must be used on both, the primary and secondary paths.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# path sec
device(config-router-mpls-path-sec)# lsp test
device(config-router-mpls-lsp-test)# to 10.1.1.100
device(config-router-mpls-lsp-test)# traffic-eng mean-rate 100
device(config-router-mpls-lsp-test)# adaptive
device(config-router-mpls-lsp-test)# soft-preemption
device(config-router-mpls-lsp-test)# secondary-path sec
device(config-router-mpls-lsp-test)# traffic-eng mean-rate 100
device(config-router-mpls-lsp-test-secpath-sec)# adaptive
device(config-router-mpls-lsp-test-secpath-sec)# soft-preemption
device(config-router-mpls-lsp-test-secpath-sec)# enable
```

Soft-preemption clean-up timer

Use the soft-preemption cleanup-timer command to set the amount of time that the point of preemption must wait to receive the Path tear notification from the ingress LSR, before sending a hard preemption path error.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enter the policy parameters.

```
device(config-router-mpls)# policy
```

4. Configure the policy parameters.

```
device(config-router-mpls-policy)# soft-preemption cleanup-timer 30
```

The following example shows setting the soft-preemption timer to 30.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# soft-preemption cleanup-timer 30
```

RASLOG messages

The following notification RASLOG messages are logged under the conditions indicated. No additional traps are generated.

1. When first path error requesting soft preemption is received for an LSP, following message is printed to RASLOG.

```
Dec  9 23:58:49 Brocade MPLS: LSP test soft preemption triggered. Preemption point 10.1.1.20
```

2. When MBB is successful for make before break setup of soft preemption requested LSP, following message is printed to RASLOG.

```
Dec  9 23:58:49 Brocade MPLS: LSP test using path NULL soft preempted with make before break
```

Path selection metric for CSPF computation

The IGP floods two metrics for every link when the MPLS traffic engineering (TE) is configured in a network. The two metrics are the IS-IS link metric and the TE link metric. To optimize the use and performance of the network, it is always better to identify specific tunnels to carry data traffic and voice traffic. This implementation allows you to specify tunnel path selection to the requirements of each type of traffic. For example, certain tunnels are to carry voice traffic (which requires low delay) and other tunnels are to carry data (where delay is acceptable).

The path calculation metric implementation allows you to specify the path calculation for a given tunnel based on either of the following requirements:

- IGP link metric for path calculation for data traffic
- TE link metric for path calculation for voice traffic

The decision of whether to use **te-metric** or **igp-metric** for CSPF computation by the LSPs is determined by CLI configurations at two levels:

- Global level: This configuration covers all RSVP LSPs (primary, secondary LSPs).
- Individual LSP level: This configuration covers all RSVP LSPs).

NOTE
The CLI configuration at the LSP level always overrides the configuration at the global level. That is, the decision to use **te-metric** or **igp-metric** for CSPF path calculation if configured at the LSP level, always overrides the configuration at the global level.

Configuring the CSPF computation mode

To configure the CSPF computation mode on a device, you must perform the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Set the cspf-computation mode under router mpls policy to use te-metric or igp-metric at the global level.

```
device(config-router-mpls)# policy
```

4. Enable or disable cspf-computation mode to use te-metric or igp-metric locally at the LSP level for primary, secondary, and bypass LSPs .

```
device(config-router-mpls-policy)# cspf-computation-mode metric-type use-igp-metric
```

The following example configures the CSPF computation mode on the device.

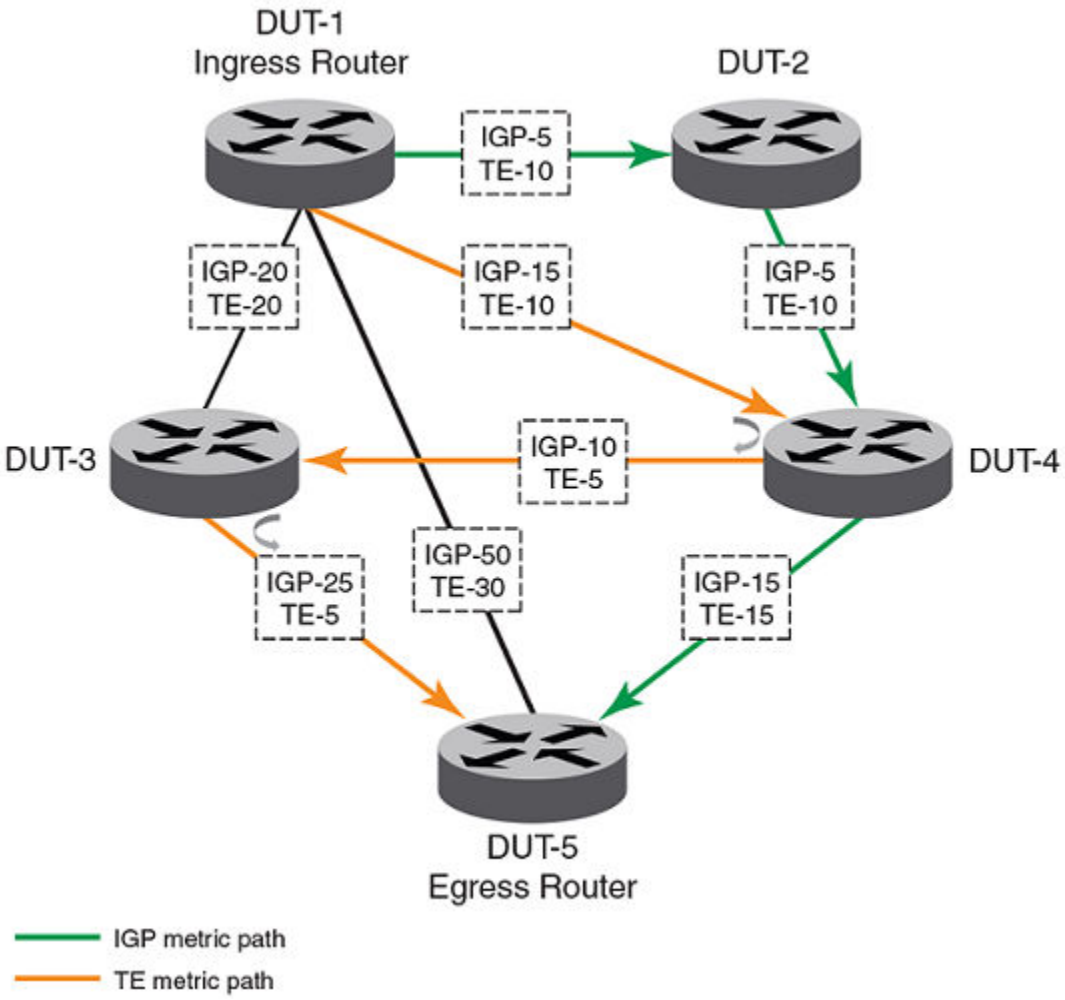
```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# cspf-computation-mode metric type use-igp-metric
```

NOTE
By default, all LSPs use global configuration.

Path selection for CSPF computation

The selection of path for the LSP depends on whether you want to use IGP or TE metric for CSPF computation. Consider the network topology in the illustration. The ingress router is DUT 1 and the egress router is DUT 5. There are two cases depending upon whether IGP metric or TE metric is used for CSPF computation.

FIGURE 10 Path selection for CSPF computation



When IGP metric is selected

The LSP selects the following path:

- DUT1 --> DUT2 --> DUT4 --> DUT5

When TE metric is selected

The LSP selects the following path:

- DUT1 --> DUT4 --> DUT3 --> DUT5

Configuring the CSPF computation mode value at global level

The user can configure the cspf-computation mode at the global level under the router mpls policy.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enter the MPLS policy mode

```
device(config-router-mpls)# policy
```

4. Configure the cspf-computation-mode.

```
device(config-router-mpls-policy)# cspf-computation-mode metric-type use igp-metric
```

In this example, the **cspf-computation-mode** is configured to use the **use-igp-metric**.

The following example show how to configure the **cspf-computation-mode** at the global level under the MPLS policy.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# cspf-computation-mode metric-type use-igp-metric
```

The user can run the **show mpls policy** command to view the configured value.

NOTE
The **use-igp-metric** or **use-te-metric** options can be enabled simultaneously.

Configuring TE-metric for an interface

You can configure the TE metric value at a specified MPLS interface level. Traffic engineering must to be enabled at the router policy level.

1. Enable the device and configure the terminal to the global configuration.

```
device# configure
```

2. Enable the MPLS router

```
device(config)# router mpls
```

3. Configure the MPLS interface level.

```
device(config-router-mpls)# mpls-interface ethernet 1/1
```

4. Enable LDP parameters.

```
device(config-mpls-if-eth-1/1)# ldp-params
```

5. Configure the te-metric value to 5.

```
device(config-mpls-if-eth-1/1-ldp-params)# te-metric 5
```

The following example shows the configuration to configure the TE-metric for an interface with a te-metric value of 5.

```
device# configure
device(config)# router mpls
device(config-mpls)# mpls-interface ethernet 1/1
device(config-mpls-if-eth-1/1)# ldp-params
device(config-mpls-if-eth-1/1-ldp-params)# te-metric 5
```

NOTE

If the te-metric uses the default value or if the **no** form of the command is used, te-metric will be equal to igp-metric value in the MPLS-TE database.

The user can run the **show mpls interface ethernet** *1/1* command to view the configured value.

Configuring TE-metric for MPLS interface

To configure TE-metric for a MPLS interface, you must perform the following steps.

1. Configure the device.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Configure the MPLS interface.

   ```
   device(config-router-mpls)# mpls-interface ethernet 1/1
   ```

4. Enable LDP parameters.

   ```
   device(config-router-mpls-if-eth-1/1)# ldp-params
   ```

5. Set the te-metric value at the MPLS interface or leave it as a default value to use the igp-metric value of the te-links for CSPF computation (optional).

   ```
   device(config-router-mpls-if-eth-1/1-ldp-params)# te-metric 5
   ```

The following example show how to configure TE-metric for a MPLS interface. In this example, the te-metric is set to 5.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 1/1
device(config-router-mpls-if-eth-1/1)# ldp-params
device(config-router-mpls-if-eth-1/1-ldp-params)# te-metric 5
```

NOTE

By default, all LSPs use global configuration.

Configuring the CSPF computation mode value for primary LSPs

The user can configure the cspf-computation-mode value at the primary LSP level.

By default, the LSP uses the global configuration at the router mpls policy. If explicitly configured, the configuration at the LSP level always overrides the configuration at the global level.

To configure the CSPF computation mode value for a primary LSP, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Set the signaled label switched path (LSP)

```
device(config-router-mpls)# lsp test
```

In this example, the LSP name is "test".

4. Configure the cspf-computation mode.

```
device(config-router-mpls-lsp-test)# cspf-computation-mode metric-type ?
```

In this step, a list of options will appear; in this case it will be **use-igp-metric** or **use-te-metric**

- `device(config-router-mpls-lsp-test)# cspf-computation-mode metric-type use-igp-metric`
- `device(config-router-mpls-policy)#no cspf-computation-mode metric-type use-te-metric`
  `Error:CSPF computation is configured to use igp-metric`
- `device(config-router- mpls-policy)#no cspf-computation-mode use-igp-metric`

The following example shows how to configure the CSPF computation mode value for the primary LSPs.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp test
device(config-router-mpls-lsp-test)#cspf-computation-mode metric-type ?
device(config-router-mpls-lsp-test)# cspf-computation-mode metric-type use-igp-metric
device(config-router-mpls-policy)#no cspf-computation-mode metric-type use-te-metric
Error:CSPF computation is configured to use igp-metric
device(config-router-mpls-policy)#no cspf-computation-mode metric-type use-igp-metric
```

In the example, the CSPF computation mode is set back to a default value of the te-metric. Run the **show mpls lsp detail** command to view the configured value.

NOTE

The configuration is not an adaptive parameter and another instance is not created when the configuration is changed without reload for adaptive LSPs but on re-optimization it takes up the new configuration to perform the cspf computation.

Global RSVP parameters

RSVP is automatically enabled when MPLS is enabled on the device. The user can optionally configure the following RSVP parameters globally at the config-router-mpls level:

- Refresh interval
- Refresh multiple

The user can also optionally configure interface-specific RSVP behaviors (RSVP authentication, RSVP reliable messaging, and RSVP refresh reduction) at the interface level.

NOTE
The effect of the **refresh-interval** and **refresh-multiple** commands can be overridden by RSVP refresh reduction behaviors.

RSVP message authentication

RSVP message authentication is implemented on the devices to prevent spoofing of RSVP messages.

RFC 2747 defines the use of a message digest carried in the RSVP INTEGRITY object. This object carries the following information:

- Key ID: An 8-bit number unique to a given sender.
- Sequence Number: A 64-bit monotonically increasing sequence number.
- Keyed Message Digest: As implemented here using MD5, it is a 16-bit message digest.

In order to support RFC 2747, this implementation supports the following:

- An authentication type using the MD5 cryptographic algorithm.
- An authentication key for use with the authentication algorithm.
- An authentication window of one (1), which specifies that the maximum number of authenticated messages that can be received out of order is one (1).

Configuring RSVP message authentication

RSVP message authentication is disabled by default. This authentication method uses MD5 and is configured within the MPLS configuration mode.

To configure RSVP message authentication, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify selected ethernet interface.

```
device(config-router-mpls)# mpls-interface ethernet 1/1
```

   In this example, the ethernet interface selected is 1/1.

4. Enable LDP parameters.

```
device(config-router-mpls-eth-1/1)# ldp-params
```

5. Enable RSVP authentication.

```
device(config-router-mpls-eth-1/1-ldp-params)# rsvp
```

6. Enable authentication.

   The prefix can be one of the following:

   • 0 = the key string is not encrypted and is in clear text.

   • 1 = the key string uses proprietary simple cryptographic 2-way algorithm.

   • 2 = the key string uses proprietary base64 cryptographic 2-way algorithm.

   ```
   device(config-router-mpls-eth-1/1-ldp-params-rsvp)# authentication key 0 administrator
   ```

   In this example, the rsvp authentication is an MD5 key named 'administrator'.

The following example shows an authentication method using MD5.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 1/1
device(config-router-mpls-eth-1/1)# ldp-params
device(config-router-mpls-eth-1/1-ldp-params)# rsvp
device(config-router-mpls-eth-1/1-ldp-params-rsvp)# authentication key 0 administrator
```

RSVP reliable messaging

The standard RSVP periodically re-sends Resv and Path refresh messages to maintain the state of the path, but many trigger messages signaling a new or changed state (such as PathTear and ResvTear messages) are sent only once. The loss of such a message can cause delays in the reservation or release of resources.

RFC 2961 provides extensions to the RSVP to make the transmission of RSVP trigger messages more reliable by creating an ID for each new RSVP message and allowing the sender to request an acknowledgment of the receipt of trigger messages.

Configuring RSVP reliable messaging

When RSVP reliable messaging is enabled on an interface of the device, RSVP trigger messages sent out on that interface includes a message ID and a request for acknowledgment from the RSVP neighbor.

When acknowledgment is not received, the trigger message is re-transmitted using the retransmission parameters configured on the interface.

NOTE
RSVP refresh messages never require acknowledgment, even when reliable messaging is enabled.

To enable RSVP reliable messaging on an interface, complete the following steps.

1. Configure the device.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable MPLS-capable interfaces for configuration and specify the individual ethernet interfaces.

   ```
   device(config-router-mpls)# mpls-interface ethernet 3/13
   ```

   In this example, the specified ethernet interface is 3/13.

4. Enable LDP parameters

```
device(config-router-mpls-eth-3/13)# ldp-params
```

5. Enable RSVP.

```
device(config-router-mpls-eth-3/13-ldp-params)# rsvp
```

6. Enable RSVP reliable messaging on the selected interface.

```
device(config-router-mpls-eth-3/13-ldp-params-rsvp)# reliable-messaging
```

The following example shows the enabling of RSVP reliable messaging on an ethernet interface 3/13.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 3/13
device(config-router-mpls-eth-3/13)# ldp-params
device(config-router-mpls-eth-3/13-ldp-params)# rsvp
device(config-router-mpls-eth-3/13-ldp-params-rsvp)# reliable-messaging
```

The previous commands enable RSVP reliable messaging on interface 3/13 with all parameters set to their defaults (or to settings previously configured on this interface, if any).

RSVP refresh reduction

RSVP control traffic (Path and Resv messages) is initially propagated to establish an RSVP session and reserve resources along the path, or to signal a change of state (trigger messages). However, because it is a soft-state protocol, RSVP also requires periodic refreshing to prevent reserved resources from aging out. The original RSVP as defined in RFC 2205 achieves this by re-sending identical Path and Resv messages (refresh messages) at regular intervals along the reserved path as long as the RSVP session remains unchanged. The bandwidth and processing time required to support these refresh messages increases linearly as more RSVP sessions are established, which can result in scaling problems.

RFC 2961 establishes extensions to RSVP which can help reduce the overhead caused by refresh messages: bundle messages, which allows multiple RSVP messages to be aggregated into a single PDU, and summary refresh messages, which replace identical RSVP message re-transmissions with a list of the IDs of all Path and Resv states to be refreshed.

When the user enables either of the refresh reduction extensions on an interface, outgoing RSVP packets sent on that interface sets the refresh reduction capability bit in the common RSVP header to indicate that the device is capable of receiving and processing refresh reduction messages and related objects.

Configuring RSVP bundle messages

When RSVP bundle messages are enabled on an interface, the device attempts to combine multiple outgoing RSVP messages on that interface into bundles to reduce overhead.

RSVP bundle messages are disabled by default for all interfaces. To enable bundle messages on an interface, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected individual ethernet interface.

```
device(config-router-mpls)# mpls-interface ethernet 3/13
```

   In this example, the specified ethernet interface is 3/13.

4. Enable LDP parameter configuration.

```
device(config-router-mpls-eth-3/13)# ldp-params
```

5. Enable RSVP configuration.

```
device(config-router-mpls-eth-3/13-ldp-params)# rsvp
```

6. Enable RSVP refresh reduction in the specified interface (3/13) with a bundle send delay of 20 milliseconds.

```
device(config-router-mpls-eth-3/13-ldp-params-rsvp)# refresh-reduction bundle-message bundle-send-
delay 20
```

   The **bundle-send-delay** option specifies the maximum period (in milliseconds) that an outgoing message can be delayed in order to create a multi-message bundle before sending. This delay is retained for the interface even when bundle messages are disabled.

The following example enables RSVP bundle messages on interface 3/13 with q bundle-send-delay of 20 milliseconds.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 3/13
device(config-router-mpls-eth-3/13)# ldp-params
device(config-router-mpls-eth-3/13-ldp-params)# rsvp
device(config-router-mpls-eth-3/13-ldp-params-rsvp)# refresh-reduction bundle-message bundle-send-delay 20
```

When the RSVP neighbor does not support refresh reduction, the interface does not bundle messages even when bundle messages are locally enabled. Use the **no** version of the command to disable RSVP bundle messages on this interface.

NOTE
Summary refresh is a more effective tool for RSVP refresh message overhead reduction.

Configuring RSVP summary refresh

When RSVP summary refresh is enabled on an interface, the device suppresses the sending of unchanged Path and Resv messages (refresh messages) and instead sends a summary message listing the IDs of Path and Resv messages that are to be refreshed.

RFC 2961 extends RSVP to create IDs for RSVP messages. Summary refresh does not affect the sending of RSVP trigger messages that signal changes of state.

RSVP summary refresh is disabled by default for all interfaces. To enable summary refresh on an interface, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces.

```
device(config-router-mpls)# mpls-interface ethernet 3/13
```

   In this example, the ethernet interface 1/3 is specified.

4. Enable LSP parameter configuration.

```
device(config-router-mpls-eth-3/13)# ldp-params
```

5. Enable RSVP configuration.

```
device(config-router-mpls-eth-3/13-ldp-params)# rsvp
```

6. Enable RSVP summary refresh reduction feature on an Ethernet interface.

```
device(config-router-mpls-eth-3/13-ldp-params-rsvp)# refresh-reduction summary-refresh
```

   This step enables the sending of RSVP summary refresh messages on interface 3/13.

The following example enables summary refresh on Ethernet interface 3/13.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 3/13
device(config-router-mpls-eth-3/13)# ldp-params
device(config-router-mpls-eth-3/13-ldp-params)# rsvp
device(config-router-mpls-eth-3/13-ldp-params-rsvp)# refresh-reduction summary-refresh
```

When the RSVP neighbor does not support refresh reduction, the interface does not send summary refresh messages, even though the feature is locally enabled. It instead continues to resend full Path and Resv refresh messages. Use the **no** version of the command to manually disable summary refresh on this interface.

Displaying refresh reduction information for an interface

The user can display RSVP refresh reduction settings for an interface by using the following command at any level of the CLI.

```
device# show mpls rsvp interface
```

RSVP message authentication on a MPLS VE interface

All inbound RSVP messages on an interface must contain a RSVP integrity object for authentication and acceptance by RSVP.

RSVP message authentication using MD5 as described in RFC 2747 is implemented on Extreme devices to prevent spoofing of RSVP messages. Inbound RSVP messages with no Integrity object, or an incorrect integrity object, is dropped by RSVP. All outbound RSVP messages on an interface contain a RSVP integrity object.

Configuring RSVP message authentication on a MPLS VE interface

RSVP Message Authentication is disabled by default. This authentication method uses MD5 for an MPLS VE interface.

Complete the following steps to configure RSVP message authentication for MPLS interface ve 100.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Specify MPLS-capable Virtual Ethernet (VE) interface.

```
device(config-router-mpls)# mpls-interface ve 100
```

   In this example, VE interface 100 is selected.

4.

```
device(config-router-mpls-if-ve-100)# rsvp
```

5. Configure RSVP message authentication for MPLS interface ve 100.

```
device(config-router-mpls-if-ve-100-rsvp)# authentication key private
```

   in this example, the variable *private* specifies a text string of up to 64 characters that is encrypted and used for RSVP message authentication.

The following example configures RSVP message authentication using MD5 for a MPLS VE interface.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ve 100
device(config-router-mpls-if-ve-100)# rsvp
device(config-router-mpls-if-ve-100-rsvp)# authentication key private
```

Displaying MPLS and RSVP information

The user can display the following information about the MPLS configuration on the device:

- Information about MPLS-enabled interfaces on the device.
- Statistics about the MPLS-enabled interfaces.
- MPLS summary information.
- Contents of the Traffic Engineering Database (TED).
- Status information about signaled LSPs configured on the device.
- Information about paths configured on the device.
- The label applied at each hop in an LSP.
- Contents of the MPLS routing table.
- RSVP information, including the status of RSVP-enabled interfaces, session information, and statistics.
- MPLS fast reroute information.

RSVP IGP synchronization

The RSVP IGP synchronization feature enables RSVP to react to an IGP neighbor down event.

RSVP IGP synchronization can help improve the convergence time of RSVP and reduce the latency in removing the resource reservations, thereby improving the overall network efficiency.

When an IGP protocol declares a neighbor down, because hello packets are no longer being received, RSVP brings down all the associated LSPs and sessions that are passing through the down neighbor. However, the IGP protocols and RSVP still act independently when bringing a neighbor up.

RSVP sessions are maintained until either the router stops receiving IGP hello packets or the RSVP Path and Resv messages time out or RSVP states are explicitly torn down by the ingress or egress. Configuring a short time for the IS-IS or OSPF hello timers allows these protocols to detect node failures quickly. Also, configuring BFD for IGP interface provides sub-second neighbor down detection time.

When quick discovery of a failed neighbor is needed, short IGP (OSPF or IS-IS) hello timers could be configured, or BFD could be enabled on IGP interfaces.

Limitations

The RSVP IGP synchronization allows RSVP to react to an IGP neighbor down event. It does not allow RSVP to detect that a neighbor node has gone down. For example, when a pair of RSVP/IGP routers are connected with parallel links, detecting one neighbor down does not mean that the entire neighbor node has gone down.

1. RSVP IGP synchronization is independent of MPLS traffic engineering configurations. Irrespective of MPLS traffic engineering configuration (OSPF or IS-IS), RSVP IGP synchronization allows MPLS (RSVP) to handle IGP neighbor down events and take action, such as tearing down the associated RSVP sessions. For example, when IS-IS is configured as MPLS-TE protocol, the user can still configure MPLS to handle an OSPF neighbor down event (and vice versa).
2. An IGP neighbor down event is handled only by the RSVP sub-component of MPLS by tearing down the associated sessions. This event is not handled by LDP sub-component of MPLS.
3. MPLS/RSVP does not keep track of the current state of IGP neighbor. That is, when an IGP neighbor goes down, RSVP tears down all the associated sessions. However, RSVP does not prevent bringing up any session while the IGP neighbor to RSVP next-hop is down (or not yet available). That is, the RSVP session is brought up even when the IGP neighbor to the next-hop does not exist.
4. An IGP neighbor down is treated as upstream neighbor down or downstream neighbor down event by RSVP, depending upon the direction of the LSP. When a downstream IGP neighbor goes down, it results in an LSP tear down or FRR switchover, whichever is applicable.
5. MPLS receives and processes an IGP neighbor down event only for the cases when an IGP neighbor goes down because of hellos not received from the peer.
6. When an IGP neighbor goes down because of an underlying interface down, MPLS does not react to an IGP neighbor down event as RSVP would also receive the interface down event and tears down associated LSPs/sessions. Handling an IGP neighbor down event is redundant in such situations.
7. When BFD is configured on IGP interfaces, an IGP neighbor down is detected quickly and may help RSVP converge faster.
8. Bypass LSPs are treated the same way as regular LSPs. Upon an IGP neighbor down, associated bypass LSPs is torn down.
9. When an IGP neighbor is Nonstop Routing or Graceful Restart (NSR/GR) capable, MPLS does not receive a neighbor down event when NSR is performed on the peer IGP router.
10. Faster FRR feature is not be triggered when MPLS detects that IGP neighbor is down. Instead, each FRR LSP is processed individually to perform local repair.
11. It is highly recommended to observe extreme caution when implementing this feature when BFD is enabled for the underlying IGP. Under some circumstances, unnecessary flapping for RSVP sessions/LSPs can occur with this combination.

Globally enabling RSVP IGP synchronization

This command globally enables the handling of an IGP neighbor down event by MPLS. This command can be executed on the fly and takes effect immediately. It is possible to enable handling of neighbor down events for IS-IS.

Configuring RSVP IGP synchronization

By default, RSVP does not handle IGP neighbor down events. RSVP IGP synchronization must be enabled to handle an IGP neighbor down event.

To configure RSVP IGP synchronization feature, the following commands need to be executed. Complete the following steps to enable RSVP to handle IGP neighbor down events for IS-IS.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable policy parameter configuration.

```
device(config-router-mpls)# policy
```

4. Configure MPLS to handle an IS-IS neighbor down event.

```
device(config-router-mpls-policy)# handle-isis-neighbor-down
```

The following example shows enabling the RSVP to handle IGP neighbor down events for IS-IS.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# handle-isis-neighbor-down
```

RSVP IGP synchronization for remote links

RSVP IGP Synchronization for remote links enables an LSP ingress router to react to neighbor down events from any location in the network. Any non-FRR can be rerouted when the router receives an IGP link or neighbor down event.

MPLS builds an IGP Sync database which is independent of the MPLS-TE database. The IGP Sync database links are keyed based on IPv4 address pair (Link IP, Router-ID of Links remote end). The link is added or updated in the database whenever an LSP Path comes UP. The IGP Link is deleted or updated whenever an LSP path goes DOWN or an IGP link down event gets generated. The IGP Link database links are associated with individual LSP instances, such that when an IGP link down event reaches MPLS, MPLS can correlate it to individual LSP paths.

An RSVP IGP synchronization remote link down event can lead to any one of the below actions:

- Bring down an LSP and retry it (setup on new path avoiding failed link) OR
- Bring down an LSP and switch to any other instance (secondary path) which is already up OR
- Create a new instance of the LSP first and then switch to the new instance.

Limitations and pre-requisites

- RSVP IGP synchronization for remote links is not usable when Traffic Engineering is not enabled in any of the routers used by the LSP.
- No additional or separate configurations are required to enable or disable.
- For full functionality of RSVP IGP synchronization for remote links, it is recommended that:
  - All routers used by all the non-FRR must be Traffic Engineering enabled.

- All transit routers shall support RRO.

Types of LSPs

Signaled LSPs

Signaled LSPs are configured at the ingress LER only. When the LSP is enabled, RSVP signaling messages travel to each LSR in the LSP, reserving resources and causing labels to be dynamically associated with interfaces. When a packet is assigned to a signaled LSP, it follows a pre-established path from the LSPs ingress LER to its egress LER. This path can be one of the following:

- A path that traverses an explicitly specified set of MPLS routers
- The IGP shortest path across the MPLS domain, determined from local routing tables
- A traffic-engineered path calculated by the device using constraints such as bandwidth reservations, administrative groups, and network topology information

Setting up signaled LSPs

An LSP consists of an actual path of MPLS routers through a network, as well as the characteristics of the path, including bandwidth allocations and routing metrics. There are two kinds of LSPs: signaled and static. Signaled LSPs are configured at the ingress LER. When the user enables a signaled LSP, RSVP causes resources to be allocated on the other routers in the LSP.

Configuring a signaled LSP consists of the following tasks:

- Specifying a path for the LSP to follow (optional)
- Setting parameters for the signaled LSP
- Specifying which packets are to be forwarded along the LSP (optional)

Setting up paths

A path is a list of router hops that specifies a route across an MPLS domain.

Once the user creates a path, the user can create signaled LSPs that sees the path. Paths are configured separately from LSPs so that a path may be specified once and then used by several LSPs that see the path by name. An LSP may specify a primary and one or more redundant paths.

A path is always configured at the ingress LER and assumes that the ingress LER is the beginning of the path. A path can contain any number of nodes, which correspond to MPLS-enabled routers in the network. Each node has one attribute: whether it is strict or loose. A strict node means that the router must be directly connected to the preceding node. A loose node means that there can be other routers in between.

Creating a path is not absolutely necessary when configuring an LSP. When the user configures a signaled LSP without naming a path, CSPF uses only information in the Traffic Engineering Database (TED), as well as the user-configured attributes and requirements of the LSP to calculate the path. When the LSP has been configured not to use CSPF, the path between the ingress and egress LERs is determined using standard hop-by-hop routing methods, as if the path consisted of a single loose node.

Complete the following steps to set up a path called *sf_to_sj* that has four nodes.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Define the explicit route path.

```
device(config-router-mpls)# path sf_to_sj
```

   In this example the explicit route path is sf to sj.

4. Configure the first hop.

```
device(config-router-mpls-path-sf_to_sj)# hop 10.150.1.1 strict
```

   In this example the IP address of the hop is 10.150.1.1.

5. Configure insert node 3.3.3.3 before 7.7.7.7.

```
device(config-router-mpls-path-sf_to_sj)# insert 3.3.3.3 strict before 7.7.7.7
```

6. Configure the loose node.

```
device(config-router-mpls-path-sf_to_sj)# hop 10.1.1.1 loose
```

7. Configure the last node.

```
device(config-router-mpls-path-sf_to_sj)# hop 10.100.1.1 strict
```

The following example shows how to set up a path called *sf_to_sj* that has four nodes.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# path sf_to_sj
device(config-router-mpls-path-sf_to_sj)# hop 10.150.1.1 strict
device(config-router-mpls-path-sf_to_sj)# insert 3.3.3.3 strict before 7.7.7.7
device(config-router-mpls-path-sf_to_sj)# hop 10.1.1.1 loose
device(config-router-mpls-path-sf_to_sj)# hop 10.100.1.1 strict
```

The path is assumed to start from the local node. The user specifies the nodes in order from ingress to egress. Specifying the local node itself as the first node in the path is optional. Further, the final node does not necessarily have to be the egress LER in the LSP. (The egress LER is specified at the LSP configuration level with the to command.) When the final node in the path differs from the egress LER, the hop between the final node in the path and the egress LER is treated as a hop to a loose node; that is, standard IP routing is used to determine the path between the final node and the egress LER.

The IP address defines an LSR and can be any interface address or a loopback interface address on the LSR.

The **strict** and **loose** parameters are relative to the preceding node. In the *sf_to_sj* path defined above, LSR 10.150.1.2 is a strict node; it must be directly connected to LSR 10.150.1.1. LSR 10.1.1.1 is a loose node; this means there can be other routers between LSR 10.150.1.2 and 10.1.1.1. When specifying a strict node, the user must make sure that the LSR is actually directly connected to the preceding node.

Modifying a path

Once the user has created a path, the user can insert or delete nodes from it.

Complete the following steps to delete a node from the *sf_to_sj* path defined in Setting up paths on page 70

1. Configure the device

```
device# configure
```

2. Enable the MPLS router.

```
device#(config)# router mpls
```

3. Identify path.

```
device(config-router-mpls)# path sf_to_sj
```

4. Delete selected path.

```
device(config-router-mpls-path)# no hop 10.1.1.1
```

   In this example, the path 10.1.1.1 is deleted.

5. Exit current level.

```
device(config-router- mpls-path)# exit
```

The following example deletes the hop 10.1.1.1.

```
device# configure
device#(config)# router mpls
device(config-router-mpls)# path sf_to_sj
device(config-router-mpls-path)# no hop 10.1.1.1
device(config-router-mpls-path)# exit
```

Inserting a hop into a path

To insert a hop into a path complete the following steps.

1. Select the path to insert the hop.

```
device(config-routr-mpls)# path sf_to_sj
```

2. Insert hop.

```
device(config-router-mpls-path-sf_to_sj)# insert 2.3.4.5 strict before 1.2.3.4
```

   The **insert** command allows a new hop to be inserted in front of an existing hop within the path. In this example, the **insert** 10.150.1.1 **strict before** 10.150.1.2 command assumes that 10.150.1.2 is already in the path and inserts 10.150.1.1 before it.

3. Exit current level.

```
device(config-router-mpls-path-sf_to_sj)# exit
```

The following example show inserting strict node 10.150.1.1 in the path before node 10.150.1.2.

```
device# configure
device(config)#router mpls
device(config-router-mpls)# path sf_to_sj
device(config-router-mpls-path-sf_to_sj)# insert 2.3.4.5 strict before 1.2.3.4
device(config-router-mpls-path-sf_to_sj)# exit
```

NOTE

When the user modifies a path, the changes are not carried over to active LSPs that see the path until the LSPs are deactivated and reactivated. For example, path *sj_to_sf* may be used by an LSP called *lsp1*. After *lsp1* has been activated, any changes to path *sj_to_sf* do not cause the route followed by *lsp1* to be modified. To get the LSP to use the modified path, the user must deactivate and then reactivate *lsp1*.

Deleting a path

To delete an entire path from the LSRs configuration, enter a command such as the following.

```
device(config-router-mpls)# no path sf_to_sj
```

Configuring signaled LSP parameters

An LSPs configuration can specify not only the path that label-switched packets follow in a network, but also the characteristics of the path, the resources allocated along the path, and actions applied to the packets by the ingress or egress LERs.

Once the user has configured a path, the user can configure signaled LSPs that see it.

The user can perform the following tasks when configuring a signaled LSP:

- Performing a Commit for an LSP
- Specifying an egress LER for the LSP
- Specifying a primary path for the LSP (optional)
- Configuring secondary or hot-standby paths for the LSP (optional)
- Setting aliases for the egress LER (optional)
- Setting a Class of Service (CoS) value for the LSP (optional)
- Allocating bandwidth to the LSP (optional)
- Configuring the setup and hold priority for the LSP (optional)
- Setting a metric for the LSP (optional)
- Including or excluding administrative groups from LSP calculations (optional)
- Limiting the number of hops the LSP can traverse (optional)
- Specifying a tie-breaker for selecting CSPF equal-cost paths (optional)
- Disabling the Record-Route function (optional)
- Disabling CSPF path calculations (optional)
- Configure Maximum Packet Size without fragmentation
- Enabling the LSP
- Disabling the LSP
- Generating Traps and Syslogs for LSPs

Resetting LSPs

The **clear mpls lsp** command allows the user to reset an RSVP LSP session. Changes in the routing table after an LSP path is established do not take effect unless the LSP is brought down and then brought up again. After the user resets the LSP, it realigns to the new routing topology. The **clear mpls lsp** command can be used on the ingress LSR of the LSP.

Resetting normal LSPs

The **clear mpls lsp** command allows the user to reset normal LSPs.

The user has the option of supplying the primary or secondary parameter for a normal LSP to reset only the primary or secondary path of the LSP.

To reset or clear a bypass RSVP LSP session complete the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Tear down then bring up the bypass LSP.

```
device(config-router-mpls)# clear mpls bypass-lsp test1
```

   In this example, the name of the LSP is *test1*.

the following example clears the bypass RSVP LSP session for LSP *test1*.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# clear mpls bypass-lsp test1
```

When the user resets an LSP with the **clear mpls lsp** command, the following information message displays:

```
"Disconnecting signaled LSP name"
"Connecting signaled LSP name"
```

Reset LSP considerations

The **clear mpls lsp** command resets and restarts an MPLS RSVP LSP.

> NOTE
> These commands are disruptive. Data traffic forwarding is impacted as the LSP is not in active state for sub-seconds after teardown. Resetting an LSP could trigger a series of actions depending upon the current state of the LSP.

The following describes the actions and state changes when an LSP is reset.

Resetting an LSP also resets the associated backup/detour LSPs:

- Resetting the primary path of an LSP causes the secondary LSP path to become active, when a hot-standby secondary path for the LSP is available. However, when the primary path comes up after the reset operation, the active path switches over from the secondary to the primary again. When the "revert-timer" is configured, the LSP path switchover may be dampened and obeys the usual revert-timer rule. There is no change in the revert-timer behavior due to the reset LSP feature.

NOTE
The above state changes are described here for informational purposes only. There could be several other intermediate state changes that are not listed here.

- Resetting the primary path of an adaptive LSP also resets the "other" new instances of the LSPs primary path, when available at the time of reset.
- Resetting the secondary path for an LSP resets the current secondary path of the LSP. It also resets the selected secondary path, when available at the time of reset.
- Resetting the secondary path for an LSP whose primary path is down may trigger the secondary path selection process to choose a new secondary path. When a new secondary path is found, it is signaled and may become the active path. When no secondary paths are found, then the current secondary may become the active path again after successful RSVP signaling.
- The primary path is UP but not active, and the secondary path is UP and active. The secondary to primary switchover occurred because the revert-timer has been configured (using a large value). Resetting the secondary LSP path still forces the path switchover from secondary to primary path in spite of the revert-timer configuration.
- For an adaptive LSP, when reset is performed before the **commit** command, then the LSP is reset and comes-up with a new set of configuration parameters. However, this is disruptive for data traffic, unlike the **commit** command, because the current instance of the LSP is reset while there is no new instance of the LSP available (because the **commit** command has not been executed yet).

MPLS bypass LSP

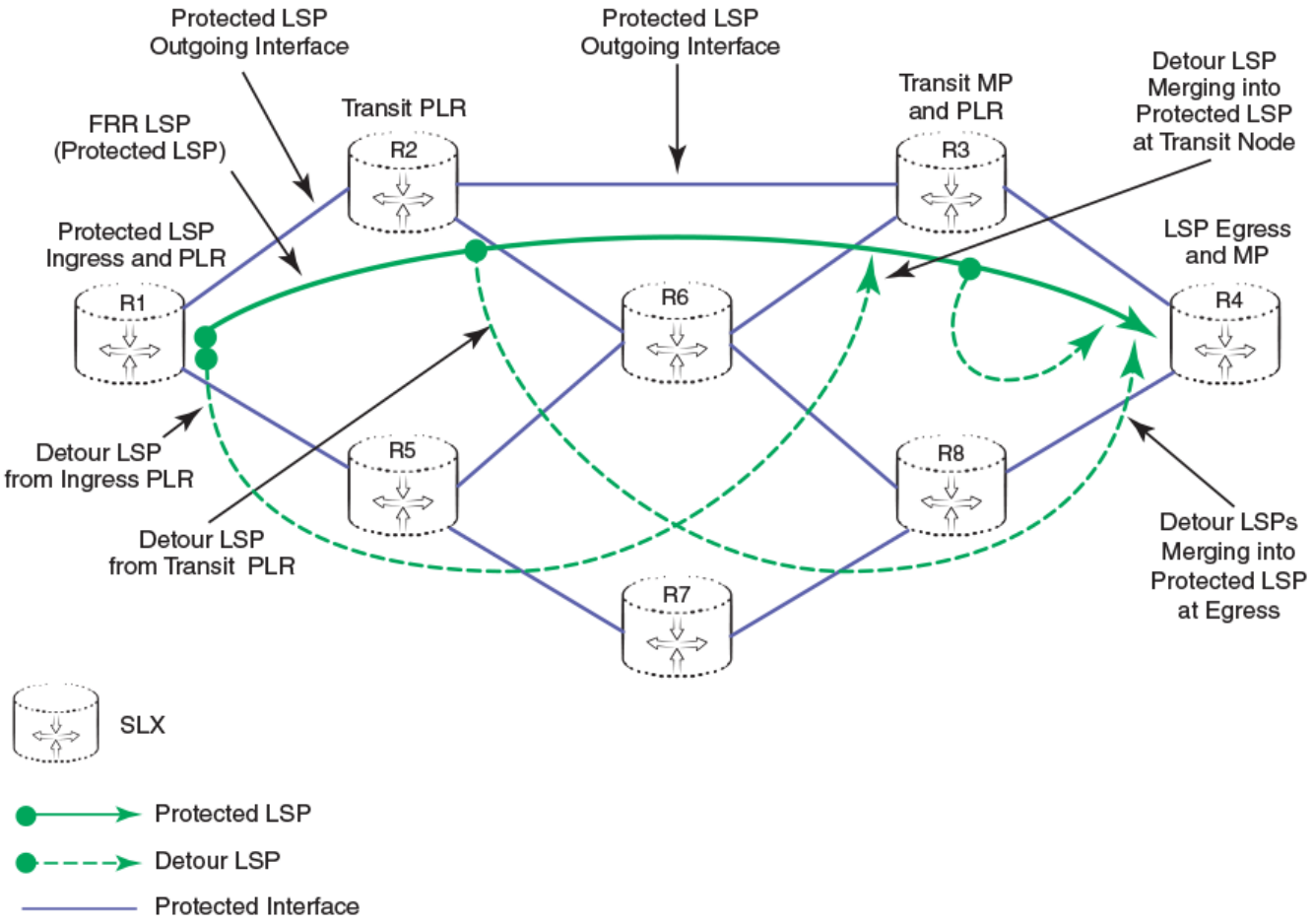
MPLS facility backup FRR

MPLS facility backup Fast Reroute (FRR) is a protection mechanism to provide fast traffic recovery, upon link or router failures. FRR LSP is also known as the protected LSP. RFC 4090 defines the FRR protection mechanism.

The FRR protection mechanism sets up detour LSPs along every node of the protected LSP so that the detour LSP carries the traffic whenever there is a failure along the outgoing interface of the protected LSP or whenever there is a failure in the downstream node of the LSP.

The Point of Local Repair (PLR) is the node of the protected LSP at which the detour LSP is set up, and the Merge Point (MP) is the node where the detour LSP merges into the protected LSP.

FIGURE 11 FRR (Protected) LSP, Detour LSP, PLR , and MP



A detour LSP at a PLR must avoid the outgoing interface of the protected LSP and merge back into any of the downstream nodes of the protected LSP. The detour must avoid any of the downstream-facing links of the protected LSP from Ingress to the PLR, avoid any nodes between the PLR and the MP, and avoid any nodes between the MP and the egress of the protected LSP.

When the detour LSP merges into the immediate downstream node of the protected LSP, it is called a link protection detour LSP. A detour LSP merges into the protected LSP at any downstream node other. The immediate downstream node is called a node protection detour LSP. A link protection detour LSP provides protection only against link failure, whereas a node protection detour LSP provides protection against link failures as well as downstream node failures.

At a PLR node, the detour LSP merging-in node or Merge Point (MP) node is chosen in the order of next-hop (NH),next-next-hop (NNHOP), next-next-next-hop (NNNHOP) and so on. This order is the default order, and explores node protection first with NNHOP MP node protection. If NNHOP MP node protection is not possible, then link protection is tried with NH as MP. If the link protection is also not possible, then NNNHOP node protection options are explored (see Figure 3).

Bypass LSP

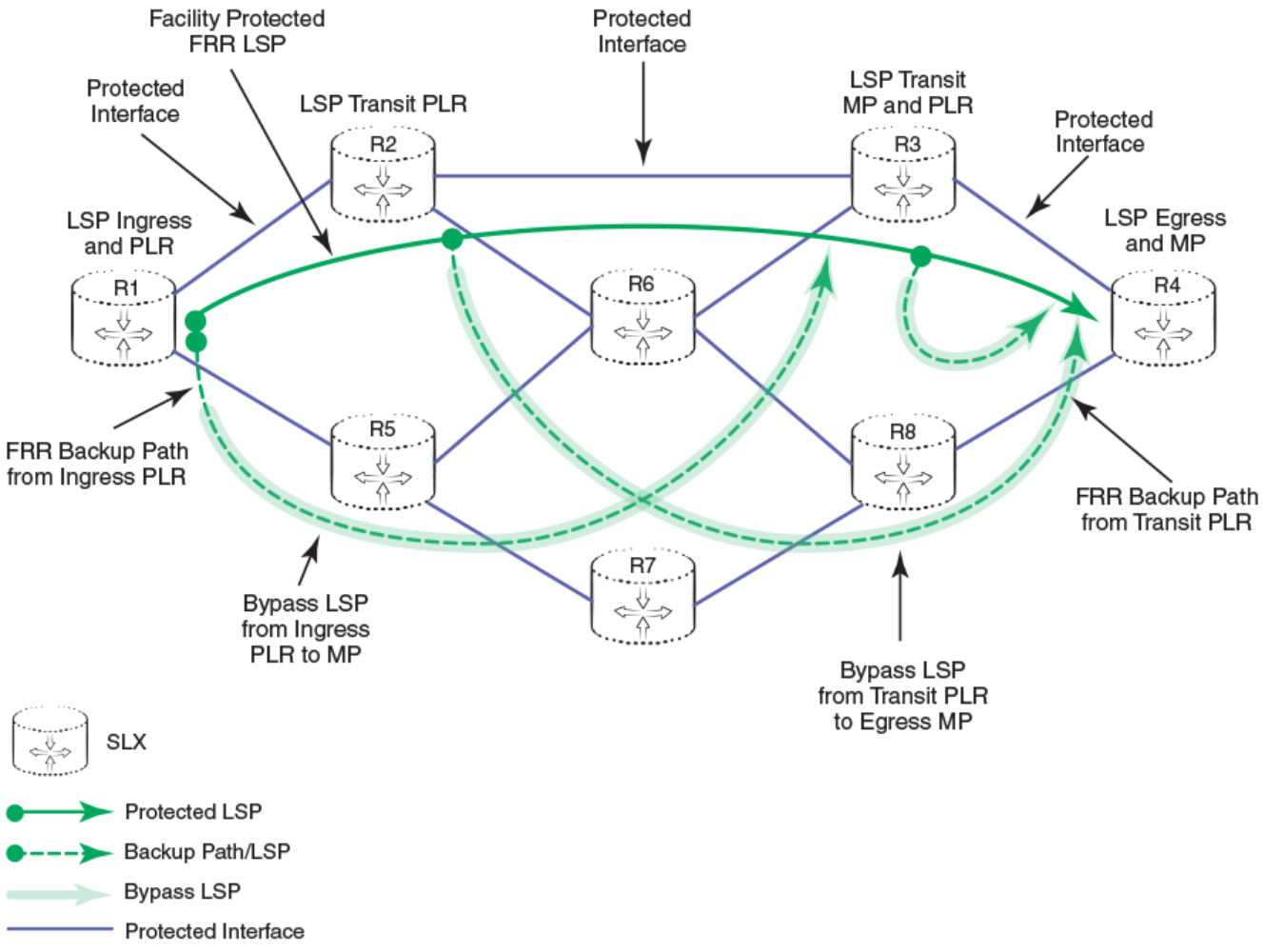
MPLS LSP tunnels that are used to provide facility-backup FRR protection are known as bypass LSPs.

The bypass LSP acts as a virtual link between the PLR and the MP nodes in the network. More than one backup LSP can use a bypass LSP, providing that all the backup LSPs have a PLR, MP, or protected interfaces in common and the bypass LSP, and satisfy the backup LSP constraints such as node and links to be avoided, backup bandwidth, backup priority, and so on.

Bypass LSPs can protect all the facility backup FRR LSPs over an MPLS outgoing interface at a PLR. The interface to which the bypass LSP provides protection is called the protected interface or the exclude interface. A bypass LSP can also provide protection to a facility-backup FRR LSP on multiple interfaces.

The bypass LSPs provide interface protection for the FRR facility-protected LSPs per RFC 4090. Bypass LSPs can protect one or more interfaces, based on the type of bypass LSP.

FIGURE 12 Facility-protected LSP, bypass LSP, and backup protection



SLX-OS supports two types of bypass LSPs, based on the way they are created:

- Static bypass LSPs
- Dynamic bypass LSPs

The user configures the static bypass LSPs similar to an LSP configuration. These bypass LSPs are operationally UP upon successful CSPF path computation and RSVP signaling. Persistence and restart behavior of the static bypass LSP are similar to regular LSPs.

Dynamic bypass LSPs are bypass LSPs that are created on demand. When a facility-protected FRR LSP requires a backup path to be up from its PLR to the MP, dynamic bypasses are created, providing that there are no existing bypass LSPs satisfying the backup path constraints. Dynamic bypass LSPs are deleted when they are unused for a certain amount of time.

The advantage of using a bypass LSP for FRR is that it improves the FRR scalability. A bypass LSP also provides a nearly hitless backup and, as a result, this improves network resiliency.

An FRR LSP is configured with a facility-backup option of the LSP configuration. This makes it a facility-backup (many-to-one) FRR LSP. The protection mechanism is node protection by default. Link protection can be explicitly configured using the **link-protection** command.

You configure the parameters of dynamically created bypass LSPs at the MPLS router mode configuration level, and the MPLS interface mode configuration levels. Dynamic bypass LSPs are deleted with a restart of the system.

A static or dynamic bypass LSP name is unique among bypass LSP names and regular LSP names in the router where the bypass LSP is being created. The bypass LSP RSVP sessions are similar to regular the LSPs.

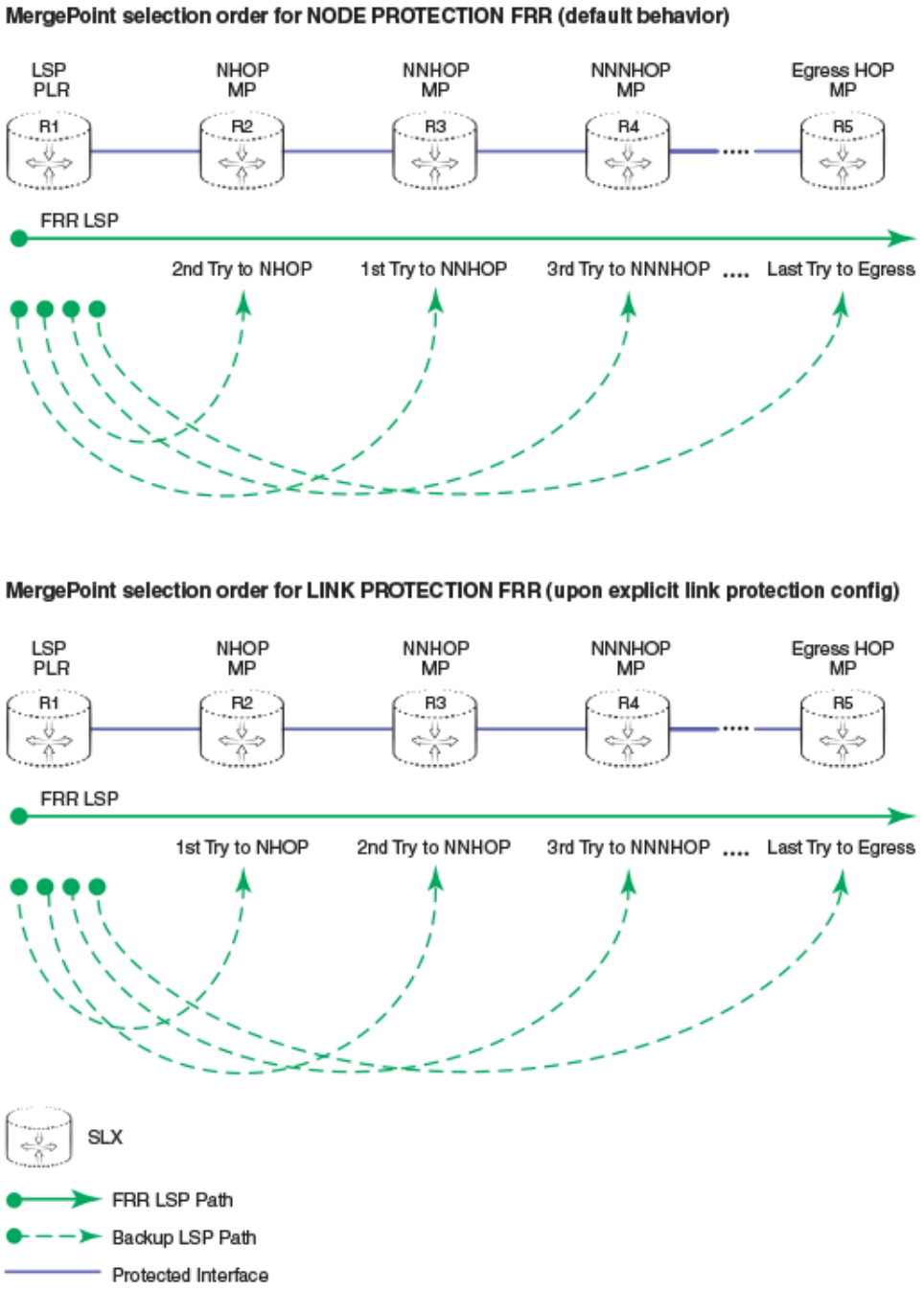
Bypass LSPs are chosen by backup LSPs as per the FRR rules:

- The bypass LSP is protecting the outgoing interface of the protected LSP.
- The bypass LSP egress node is the Merge Point (MP) node of the requested backup.
- The bypass LSP is not using any of the FRR LSP links (downstream) from the ingress to the PLR.
- The bypass LSP is not using any of the FRR LSP nodes downstream between the PLR and the MP.
- The bypass LSP not using any of the FRR LSP nodes downstream between the MP and the LSP egress.
- The bypass LSP satisfies backup constraints such as the priority, bandwidth, exclude interface, exclude nodes, and so on.

The link protection FRR facility-protected LSP selects its MP in the order of next hop (NHOP), next-next hop (NNHOP), and so on until it reaches the egress node.

The node protection FRR facility protected LSP selects its MP in the order of next-next hop (NNHOP), next hop (NHOP), next-next-next hop (NNNHOP) and so on until it reaches the egress node.

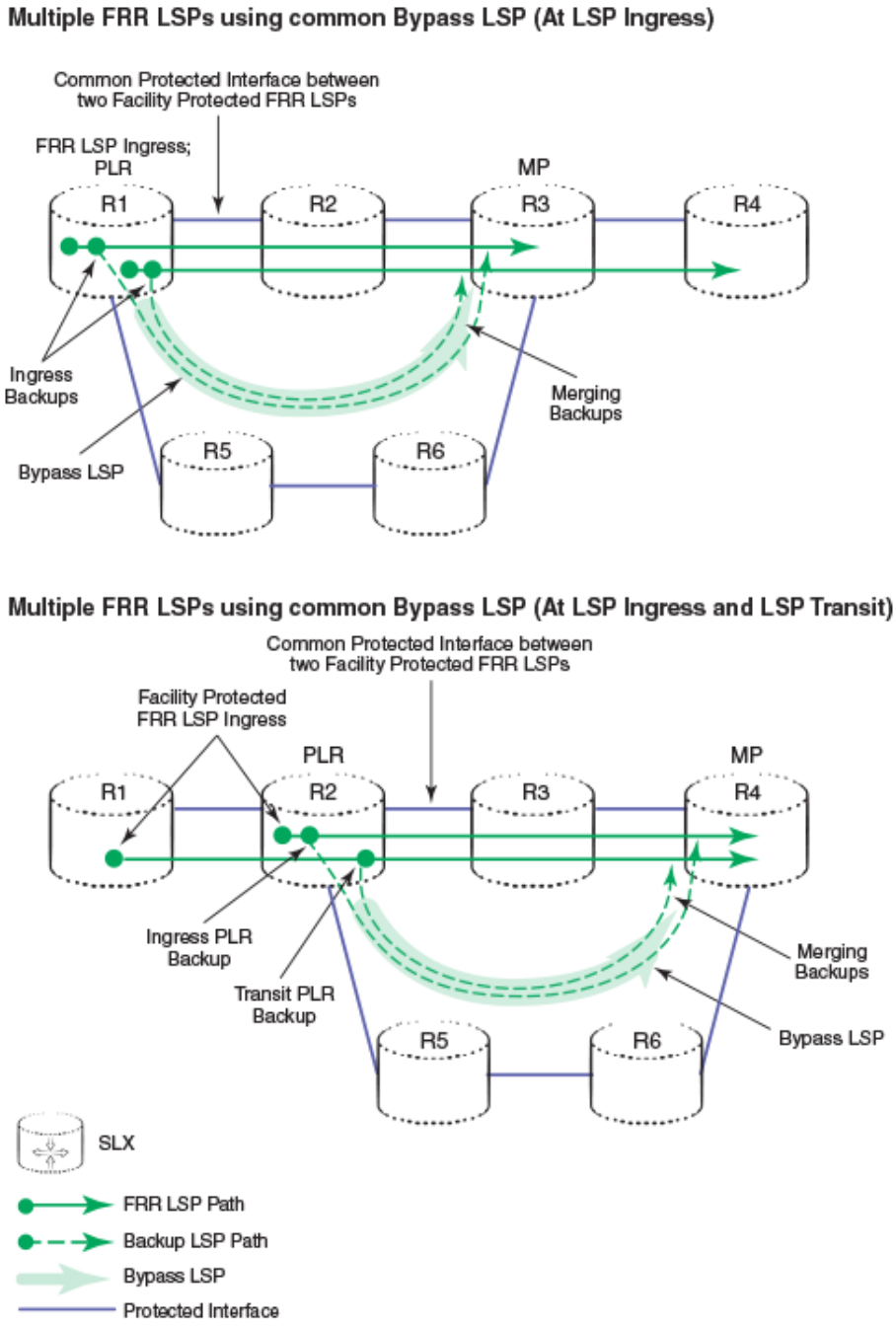
FIGURE 13 Merge Point selection order for node protection and link protection FRR



The bypass LSP protects at least one interface. Such an interface is known as an exclude interface or a protected interface. There can be multiple bypass LSPs protecting an exclude- interface.

All facility protected LSPs outgoing through the exclude-interface can establish their backup LSP sessions on the bypass LSP provided that the bypass LSP reaches the same MP required by the FRR LSP backup LSPs, and the bypass LSP satisfies backup path and session constraints.

FIGURE 14 Multiple FRR LSPs sharing a protected interface and a common bypass LSP



When an exclude interface goes down, all facility protected LSPs (outgoing through the exclude interface) and backup LSPs become active over the bypass LSP and start passing protected LSP traffic (along the PLR and the MP) over the bypass LSP.

Adaptive bypass LSP

Bypass LSPs can be adaptive or non-adaptive, based on a configuration similar to regular LSPs.

When a bypass LSP is adaptive, it allows for modification to some of its parameters while enabled. Without the adaptive capability, the user must disable the bypass LSP before modifying its parameters.

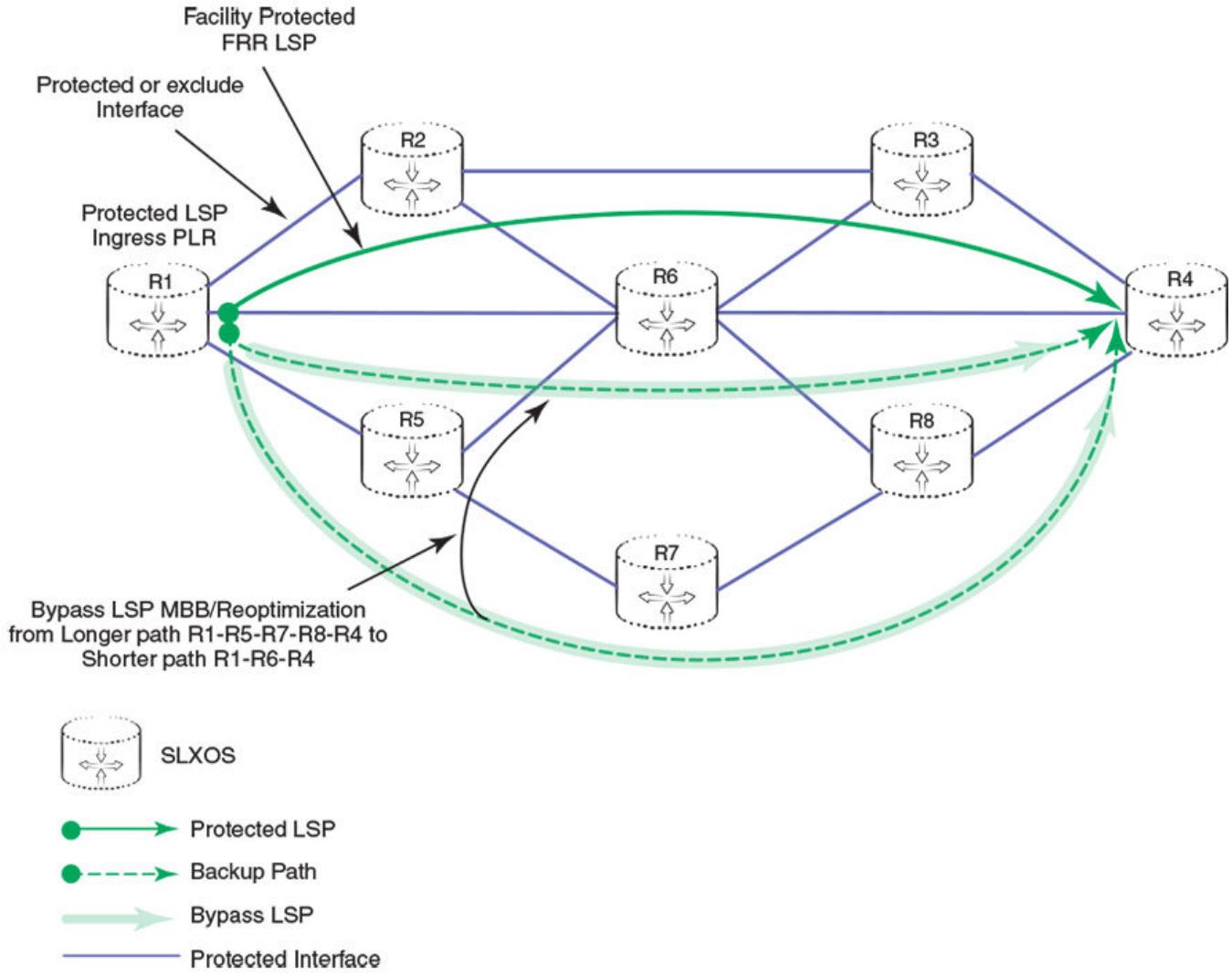
LSP make-before-break (MMB) is a procedure where a new instance of the LSP is brought up, and a switch changes from the current instance to the new instance of the LSP. The new instance of the LSP may have been setup with one or both of the following characteristics:

- One or more modified LSP parameters.
- Newly calculated LSP path. The new calculated LSP path may be a better path compared to the current instance.

Modifying any adaptive parameters of an adaptive bypass LSP creates a new instance of the bypass LSP with the updated parameter. Upon automatic or manual commit of changes to the bypass LSP, the new instance of the bypass LSP calculates a new path using newly modified parameters and tries to bring itself up. Once the new instance is up, a make-before-break (MBB) switch takes place from the current instance of the bypass LSP to the new instance.

Below is an example of a bypass LSP re-optimization at an FRR LSP ingress.

FIGURE 15 Bypass LSP re-optimization at FRR LSP ingress



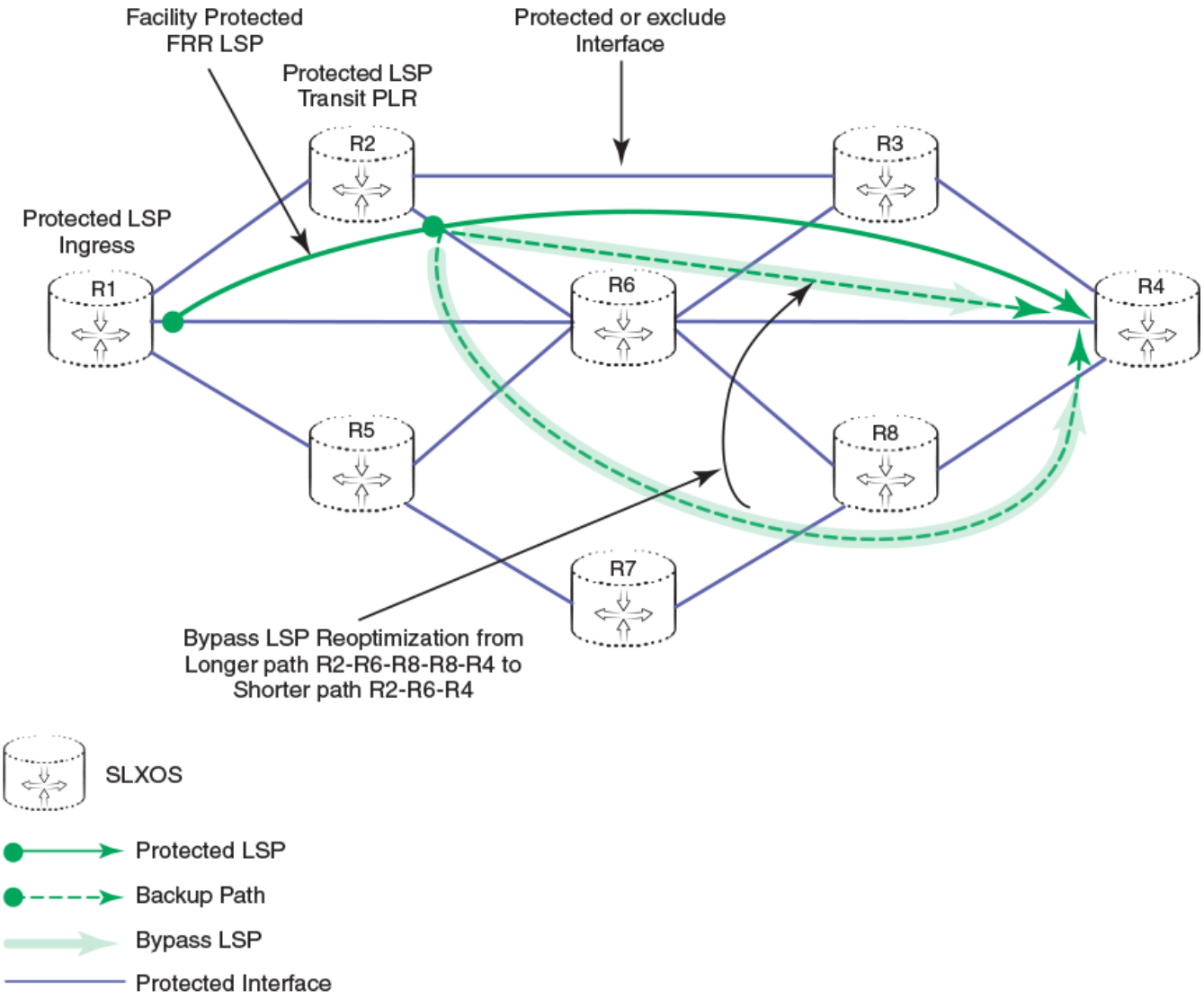
Adding the adaptive capability to a bypass LSP enables the bypass LSP to switch to a better route by way of an action known as LSP path re-optimization. The calculation of the re-optimized route can occur on demand based on user request or periodically based on the expiration of a configurable re-optimization timer. Path re-optimization succeeds when a better path becomes available, a new instance of a better path comes up, and an MBB switch takes place from the current instance to the new instance.

An adaptive bypass LSP MBB new instance does not take place if there are any active (traffic carrying or backup in a switched state) backups on the bypass LSP.

When a bypass LSP goes through MBB due to re-optimization, or an adaptive parameter change, all backups using it are torn down from that bypass. After this, the FRR LSP triggers a new backup query. This new query procedure is independent of the previously associated bypass LSP of the backup path. Based on the network topology, availability, and constraint fulfilment, the same old bypass LSP may or may not be chosen.

Below is an example for bypass LSP MBB due to re-optimization or a new instance commit.

FIGURE 16 Bypass LSP re-optimization at FRR LSP transit PLR



Best bypass LSP selection

Multiple bypass LSPs can satisfy a backup path request to a MP. When both static bypass LSPs and dynamic bypass LSPs are available to an MP, the statically created bypass LSPs are preferred over the dynamically created bypass LSPs.

When there are multiple bypass LSPs of the same type, selection among them is based on the bypass LSP path cost and the number of backups currently associated with the bypass LSP. The first level of selection is based on the bypass LSP path cost. When there are multiple equal-cost bypass LSPs, then the bypass LSP with least number of riding backup paths is chosen.

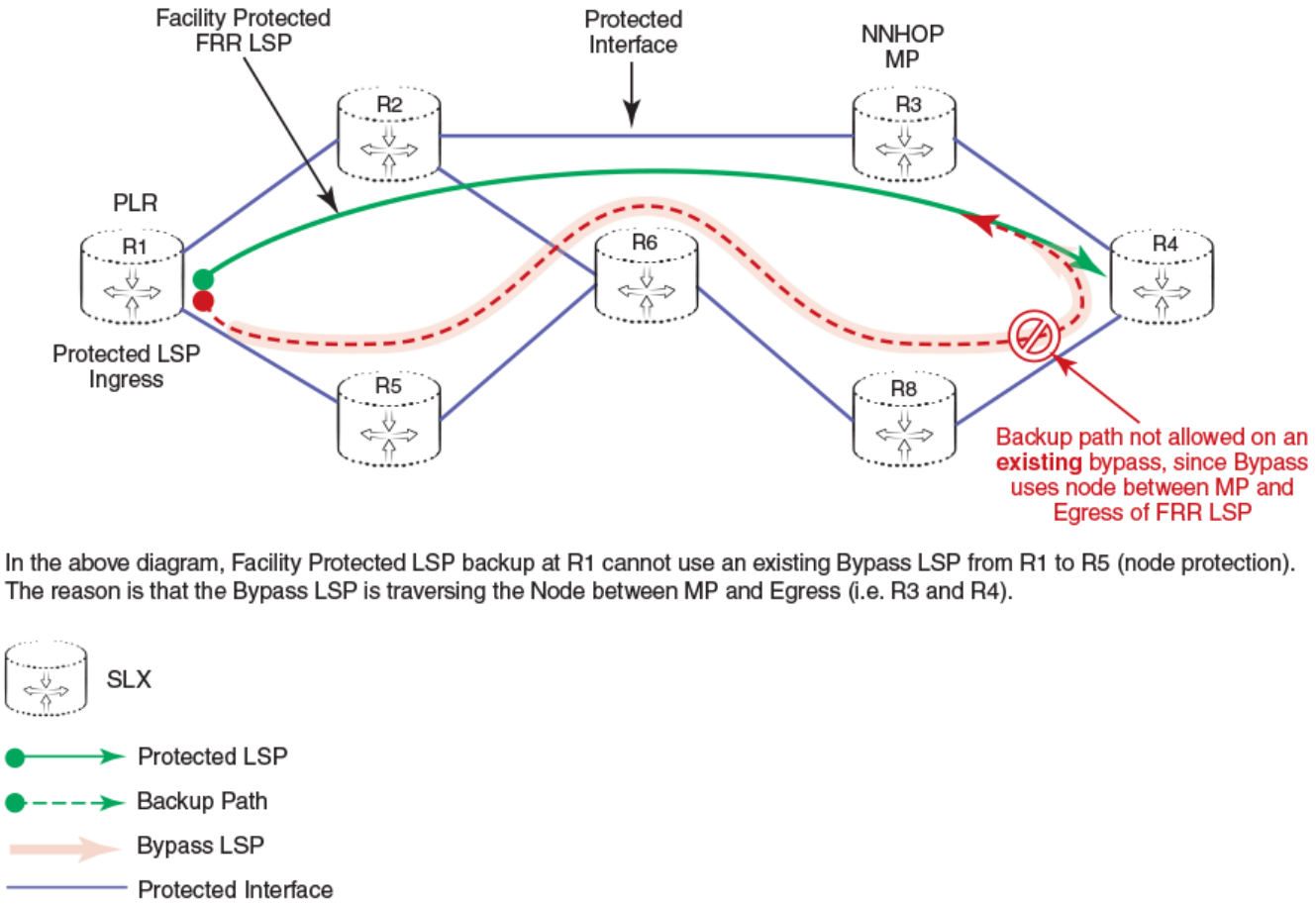
By default, while selecting a bypass LSP for backup, the cost of a bypass LSP is considered as zero (0). In such a scenario, only the number of riding backups is used for selecting between equal zero-cost bypass LSPs.

Both the cost and the number of riding backups are considered when selecting a bypass LSP using the **cspf-computation-mode use-bypass-metric** command in the MPLS router policy configuration mode.

Liberal bypass LSP selection

Fast Reroute (FRR) rules state that if an LSP is traversing the nodes between the MP and the egress of a protected LSP, then the FRR LSP does not chose a bypass LSP for its backup path.

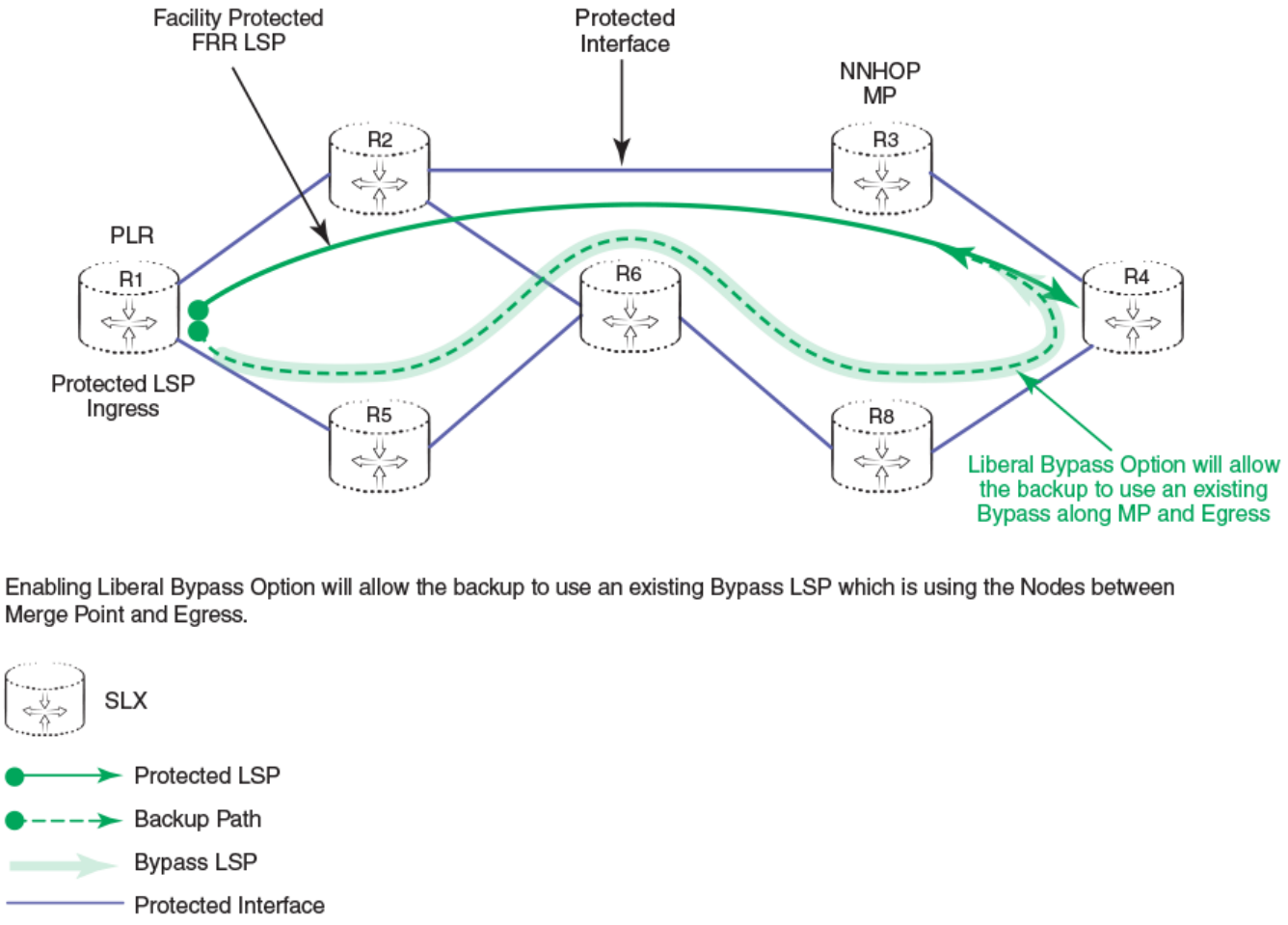
FIGURE 17 Regular bypass LSP selection



MPLS liberal bypass FRR protection is a special case where a rule of selecting a bypass LSP is liberalized and is allowed to select a bypass LSP even if it is traversing the FRR LSP downstream nodes between the MP and the egress.

The liberal bypass option reduces the number of bypasses, either dynamic or static, to create with the network.

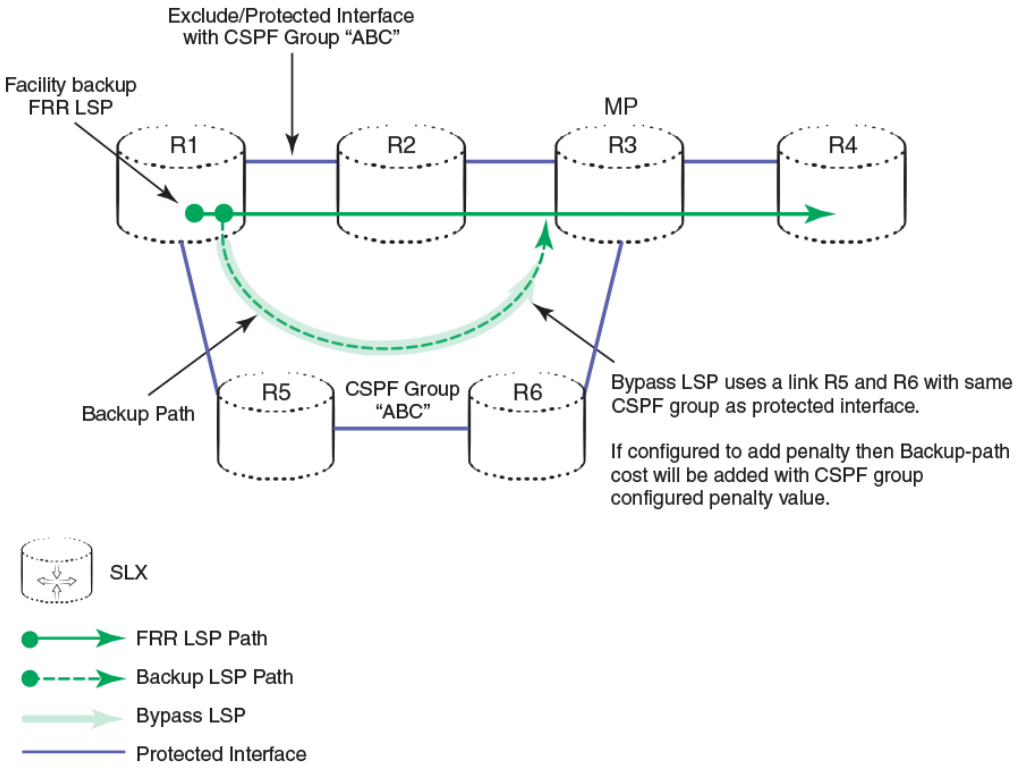
FIGURE 18 Liberal bypass LSP selection



CSPF group penalty for backup path

MPLS interfaces or nodes used by the bypass LSP may be part of a CSPF group. Therefore, a backup path must avoid any bypass LSPs that are using the same CSPF group as the protected interface. If the chosen bypass LSP uses the same links that are of same CSPF group as the protected interface or exclude interface, then a penalty can be added to the backup path cost.

FIGURE 19 Backup path bypass using the same CSPF group as a protected interface



FRR LSP switch to backup path on bypass LSP

Facility-protected FRR LSP uses a bypass LSP to carry its traffic whenever there is a failure in its protected path. Multiple events can lead to switching traffic from a protected LSP to a backup path. Some primary reasons for switching the protected LSP traffic to its corresponding backup paths over a bypass LSP include a protected or excluded interface going down, or a NHOP node going down.

At every PLR node of a facility-protected LSP, whenever a backup path is associated with a bypass LSP, the following actions take place:

- The protected LSP shows the bypass selection in its FRR details section. The protection status shows as up.
- The bypass LSP shows the backup path session information. The backup path status shows as up. The backup path role shows as Ingress if it is a protected LSP ingress PLR; otherwise, the backup path role shows as Transit.
- An Ingress Backup (BI) RSVP session is created corresponding to the backup path. There is no corresponding Merged Backup (BM) or Egress Backup (BE) RSVP session at the MP. BM and BE sessions are created upon the FRR switch.
- There are no RSVP path or reservation messages corresponding to a backup traversing over the bypass LSP before the FRR switch.

The facility-protected FRR LSP initiates the FRR switch based on the FRR switch trigger event. Upon FRR protection switch trigger, the following actions take place:

- The protected path data traffic is switched to backup path over the bypass LSP. At the PLR, switched traffic has a single backup label if the bypass LSP is a single hop and the MP is in transit. There is no label if the bypass LSP is a single hop and the MP is a protected LSP egress. There is a single bypass label if the bypass LSP is multi-hop and the MP is a protected LSP egress. There are two labels if the bypass is multi-hop and the MP is in transit.
- The backup session path message to the MP is initiated to bring up the MP node Merged Backup (BM) or Egress Backup (BE) RSVP session.

- The backup session RSVP path message is carried over the bypass LSP.
- The backup session RSVP reservation message follows any possible return path (from MP to PLR) in the network, per IP routing.
- A protected LSP shows the backup status as Active.
- A bypass LSP shows the backup status as Active.
- The bypass LSP MBB or re-optimization is not allowed until the backup status is active.

At any time, it is possible that there can be Active, switched, or up, as well as non-switched backup paths on the bypass LSP.

Configuring a static bypass LSP

Configure the bypass LSP by specifying a unique name. The name must be unique across the regular LSPs and the bypass LSPs in the system. The **bypass-lsp** command creates an instance of the bypass LSP with its administrative status as down. You must configure a few of the mandatory parameters in order to enable (admin up) the bypass LSP. You can delete the bypass LSP by using the **no bypass-lsp** form of the command.

To configure the static bypass LSP, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name of my-bypass-lsp.

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

The following example shows how to configure a static bypass LSP with a unique bypass LSP of my-bypass-lsp.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)#
```

Enabling and disabling a bypass LSP

You can enable (admin up) a bypass LSP or disable (admin down) a bypass LSP using the **enable** and **no enable** commands respectively. You must configure the bypass LSP with the **to-address** and **exclude-interface** commands before enabling the **bypass-lsp** command.

To enable a bypass LSP, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with a unique static bypass LSP name such as my-bypass-lsp.

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Enable of disable the bypass LSP.
   - To enable the bypass LSP, use the following command.

     ```
     device(config-router-mpls-bypass-lsp-my-bypass-lsp)# enable
     ```

   - To disable the bypass LSP use the following command.

     ```
     device(config-router-mpls-bypass-lsp-my-bypass-lsp)# no enable
     ```

The following example shows how to enable the bypass LSP.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# enable
```

Configuring a bypass LSP adaptive parameter

Bypass LSPs can be made adaptive by using the **adaptive** command. By default, bypass LSPs are non-adaptive . The user can modify the adaptive bypass LSP adaptive-parameters without disabling the LSP. When the LSP is UP, then modifying an adaptive parameter, such as exclude-interface or bandwidth, leads to the creation of a new instance of the bypass LSP. Adaptiveness can be manually enabled by using the **adaptive** command and can be disabled by **no adaptive** for of the same command.

To make the bypass LSP adaptive, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Configure the bypass LSP with the unique static bypass LSP name of *my-bypass-lsp* .

   ```
   device(config-router-mpls)# bypass-lsp my-bypass-lsp
   ```

4. Configure the command to be adaptive.

   ```
   device(config-router-mpls-bypass-lsp-my-bypass-lsp)# adaptive
   ```

The following example combines the steps above to configure a static bypass LSP as adaptive with a unique bypass LSP name of *my-bypass-lsp* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# adaptive
```

Configuring bypass LSP exclude interface

At least one excluded interface is mandatory to enable the bypass LSP. Configure static bypass LSPs with at least one exclude-interface or protected interface.

The user must decide on which interface to be protected when that interface is being used by the facility protected FRR LSPs. When an exclude interface goes down, all the facility protected LSP backup paths on the bypass become active and start passing traffic on the bypass LSP.

The user can configure more than one exclude-interface in a bypass LSP. In such a case, the bypass LSP can protect backup sessions from all facility protected FRR LSPs on all the exclude interfaces, provided that their MP and bypass egress match.

An MPLS interface which is configured as an exclude interface for a static bypass LSP is not allowed to be manually unconfigured.

To configure a bypass LSP with an excluded interface, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name of my-bypass-lsp.

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the static bypass to exclude selected interfaces.

```
device(config-router-mpls-my-bypass-lsp)# exclude-interface Ethernet 2/8
device(config-router-mpls-my-bypass-lsp)# exclude-interface ve 203
device(config-router-mpls-my-bypass-lsp)# exclude-interface Ethernet 2/6
```

The following example combines the steps above to configure a static bypass LSP to exclude selected interfaces with a unique bypass LSP name of *my-bypass-lsp* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-my-bypass-lsp)# exclude-interface Ethernet 2/8
device(config-router-mpls-my-bypass-lsp)# exclude-interface ve 203
device(config-router-mpls-my-bypass-lsp)# exclude-interface Ethernet 2/6
```

Configuring bypass LSP to address

A bypass LSP must be configured with destination IPv4 to address. The address must be the IP address of any of interface on the router which is reachable from the ingress router. The IP address containing router becomes a merge point for any of the backup paths using the bypass LSP.

To configure a bypass LSP to address, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name of *my-bypass-lsp* .

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the bypass LSP with the destination IPv4 address of *10.20.1.20* .

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# to 10.20.1.20
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP name of *my-bypass-lsp* and a destination IPv4 address of*10.20.1.20* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# to 10.20.1.20
```

Configuring bypass LSP from address

The user can configure a bypass LSP with an optional from IPv4 address. When configuring, use this IP address as a from IP address in the RSVP session. The from address is an adaptive parameter.

To configure a bypass LSP from address, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Configure the bypass LSP with the unique static bypass LSP name of *my-bypass-lsp* .

   ```
   device(config-router-mpls)# bypass-lsp my-bypass-lsp
   ```

4. Configure the bypass LSP **from** address.

   ```
   device(config-router-mpls-bypass-lsp-my-bypass-lsp)# from 10.20.1.10
   ```

   In this scenario, the selected from address is *10.20.1.10* .

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP name of *my-bypass-lsp* and an IPv4 from address of *10.20.1.20* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# from 10.20.1.10
```

Configuring a bypass LSP record route

The user can configure a bypass LSP with a record route option for RSVP session Route recording. The record route option enables by default. The user can manually enable or disable the command by using the **record enable** and **record disable** commands respectively.

To enable or disable the bypass LSP record route, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Configure the bypass LSP with the unique static bypass LSP name of *my-bypass-lsp* .

   ```
   device(config-router-mpls)# bypass-lsp my-bypass-lsp
   ```

4. Enable or disable the record command.
   - To enable the record command.

     ```
     device(config-router-mpls-bypass-lsp-my-bypass-lsp)# record enable
     ```

   - To disable the record command.

     ```
     device(config-router-mpls-bypass-lsp-my-bypass-lsp)# record disable
     ```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP name of *my-bypass-lsp* and a record reroute option of **record enable** or **record disable**.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# record enable
-or-
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# record disable
```

Configuring a bypass LSP CSPF computation mode

The user can base a bypass LSP CSPF computation on an IGP metric or a TE metric. By default, the CSPF computation mode sets to the TE metric. The user can set the CSPF computation mode by using the **cspf-computation-mode** command and its options. This parameter is an adaptive parameter.

To configure a bypass LSP CSPF computation mode, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Configure the bypass LSP with the unique static bypass LSP name of *my-bypass-lsp* .

   ```
   device(config-router-mpls)# bypass-lsp my-bypass-lsp
   ```

4. Configure the **cspf-computation-mode** command with the **use-igp-metric** option or the **use-te-metric** option.
   - Configuring the **cspf-computation-mode** command with the **use-igp-metric** option.

     ```
     device(config-router-mpls-bypass-lsp-my-bypass-lsp)# cspf-computation-mode  use-igp-metric
     ```

   - Configuring the **cspf-computation-mode** command with the **use-te-metric** option.

     ```
     device(config-router-mpls-bypass-lsp-my-bypass-lsp)# cspf-computation-mode  use-te-metric
     ```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP name of *my-bypass-lsp* and a **cspf-computation-mode** with the **use-igp-metric** or **use-te-metric** option.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# cspf-computation-mode  use-igp-metric
-or-
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# cspf-computation-mode  use-te-metric
```

Configuring a bypass LSP CSPF tie breaking option

A bypass LSP CSPF computation can use tie breaking options when there are multiple equal cost paths with equal hops. The tie breaking option can be random, most fill, or least fill. The use can configure these options using the **tie-breaking** command. Consider the **random** option as the default option. This parameter is an adaptive parameter.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name of *my-bypass-lsp* .

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the **tie-breaking** command with one of the following choices.
   - Configure the **tie-breaking** command with the **random** option.

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# tie-breaking random
```

   - Configure the **tie-breaking** command with the **most-fill** option.

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# tie-breaking most-fill
```

   - Configure the **tie-breaking** command with the **least-fill** option.

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# tie-breaking least-fill
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP name of *my-bypass-lsp* and a choice of three different **tie-breaking** command options.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# tie-breaking random
-or-
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# tie-breaking most-fill
-or-
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# tie-breaking least-fill
```

Configuring bypass LSP CoS parameter

The user can configure the bypass LSP Class of Service (CoS) value using the **cos** command. By default, the Cos parameter is un-configured. Cos parameter is a non-adaptive parameter and requires the bypass LSP to be in an administrative DOWN state during modification.

To configure the bypass LSP CoS parameter, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name. In this example, the name is *my-bypass-lsp* .

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the Class of Service (Cos). In this example, the chosen value is five (*5*).

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# cos 5
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP of *my-bypass-lsp* and a CoS value of five (*5*).

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# cos 5
```

Configuring a bypass LSP hop limit

The user can set the bypass LSP hop limit by using the **hop-limit** command. The hop limit is an adaptive parameter. The hop limit, by default, is not configured.

To configure the bypass LSP hop limit , complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name. In this example, the name is *my-bypass-lsp* .

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the hop limit. In this example, the hop limit configuration is *10* .

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# hop-limit 10
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP of *my-bypass-lsp* and the hop limit is *10* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# hop-limit 10
```

Configuring bypass LSP priority values

The user can configure the bypass LSP setup and hold priority values by using the **priority** command. Only those backup paths with a priority less than or equal to the holding-priority of the bypass LSP are able to ride the bypass LSP.

To configure the bypass LSP priority values, complete the following steps.

1. Enable the device foe configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name. In this example, the name is *my-bypass-lsp*.

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the **priority** command. In this example, the **set-up-priority** is configured to six (*6*) and the **hold-priority** is set to four (*4*).

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# priority 6 4
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP of *my-bypass-lsp* and priority values of six (*6*) for the **setup-priority** and a value of four (*4*) for the **hold-priority** .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# priority 6 4
```

Configuring bypass LSP bandwidth parameters

The user can configure the bypass LSP bandwidth parameters using the **traffic-engineering** command. There are three parameters, **max-burst**, **max-rate**, and **mean-rate**. All the three traffic-engineering parameters are adaptive parameters.

The bandwidth protected backup paths consume the bypass bandwidth.

to configure the bypass LSP bandwidth parameters, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name. In this example, the name is *my-bypass-lsp* .

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the bypass LSP bandwidth parameters. In this example, the **mean-rate** is configured to *2000* kbps., the **max-rate** is configured to *4000* kbps., and the **max-burst** is configured to *3000* kbps.

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# traffic-engineering mean-rate 2000 max-rate
4000 max-burst 3000
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP of my-bypass-lsp with the **mean-rate** configured to *2000* kbps., the **max-rate** configured to *4000* kbps., and the **max-burst** configured to *3000* kbps.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# traffic-engineering mean-rate 2000 max-rate 4000 max-
burst 3000
```

Configuring a bypass LSP re-optimization timer

The user can configure a bypass LSP re-optimization timer on an adaptive bypass using the **reoptimize-timer** command.

To configure the re-optimization timer, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name of *my-bypass-lsp* .

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the re-optimization timer. In this example, the timer is set to 400 seconds.

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# reoptimize-timer  400
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP name of *my-bypass-lsp* and configures the re-optimization timer to 400 seconds.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# reoptimize-timer  400
```

Configuring a bypass LSP primary path

The user can configure a bypass LSP with an explicit path using the **primary-path** command. The path can contain strict and loose hops. Use this path for calculating the path of the bypass LSP, similar to regular LSPs. The path must be defined before configuring it into a bypass LSP. the primary path is an adaptive parameter. By default, there is no configuration of the primary path.

To configure a bypass LSP primary path, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name. In this example, the name is *my-bypass-lsp* .

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Configure the primary path. In this example, the selected primary path is called *my-bypass-path* .

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# primary-path  my-bypass-path
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP of *my-bypass-lsp* and configures a primary path with the name *my-bypass-path* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# primary-path  my-bypass-path
```

Committing adaptive parameters of a bypass LSP

The user can modify the bypass LSP parameters when the configuration is adaptive. When the bypass LSP is operationally UP, then any modification of the adaptive parameters of the bypass LSP creates a new instance of the bypass LSP if it is not present. Once modifying one or more adaptive parameters, the user can commit the changes so the new instance of the bypass LSP can be try to bring it to the UP status. When the new instance of the bypass LSP is successfully UP, then a make before break switch happens from current instance of the bypass to the newly brought up instance.

To commit adaptive parameters of a bypass LSP, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the bypass LSP with the unique static bypass LSP name of *my-bypass-lsp* .

```
device(config-router-mpls)# bypass-lsp my-bypass-lsp
```

4. Enable the **commit** command.

```
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# commit
```

The following example combines the steps above to configure a static bypass LSP with a unique bypass LSP name of *my-bypass-lsp* and the **commit** command is enabled.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# bypass-lsp my-bypass-lsp
device(config-router-mpls-bypass-lsp-my-bypass-lsp)# commit
```

Dynamic bypass LSP

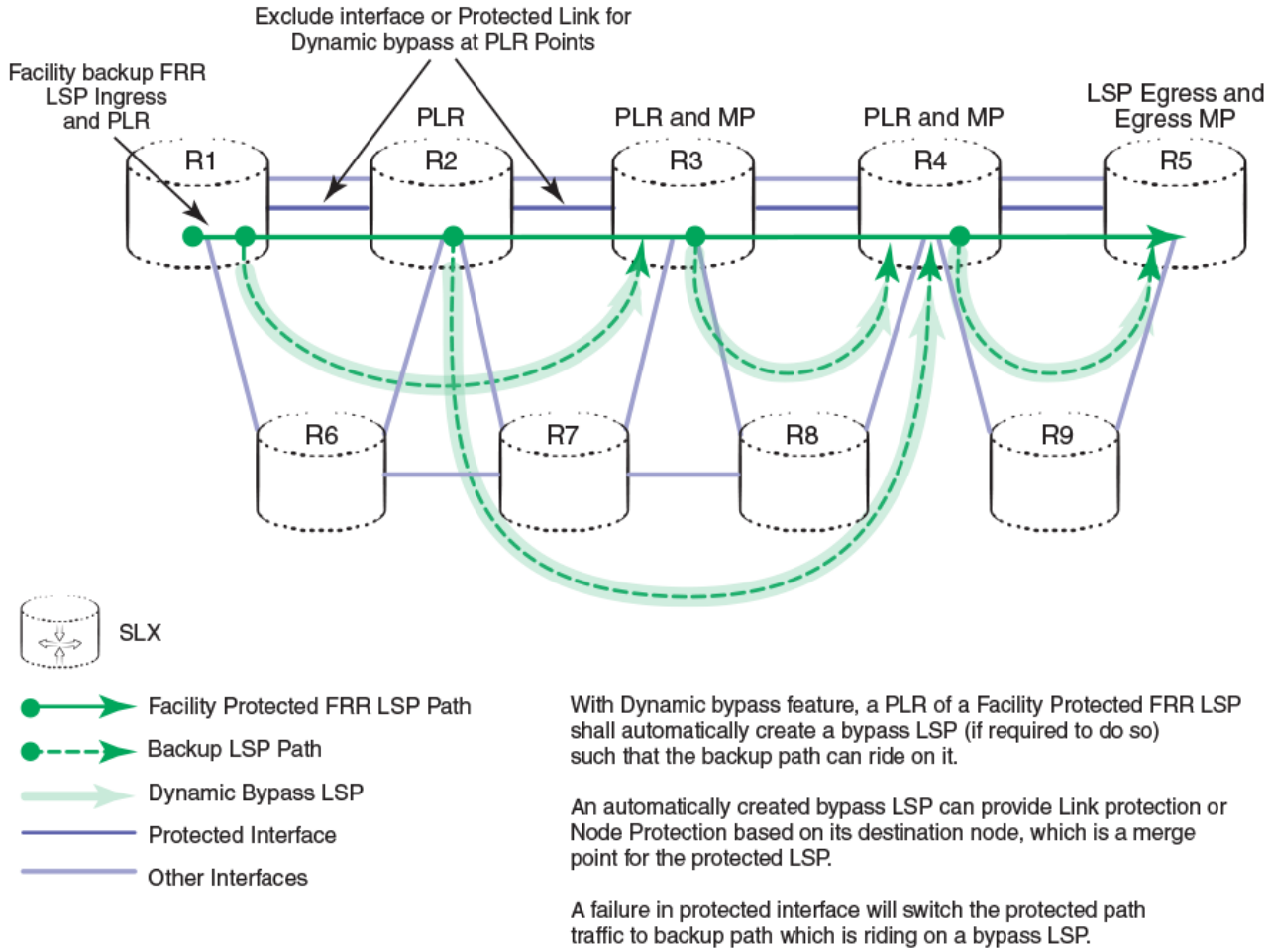
A dynamic bypass references a feature which enables the automatic creation of a bypass LSP.

When the router or system auto-creates a bypass LSP at runtime, these bypass LSPs are called dynamic bypass LSPs. Dynamic bypass LSPs are created at a FRR LSP PLR when there is a requirement to provide FRR facility backup protection. At every PLR, the facility protected FRR LSP requiring backup can trigger the creation of a dynamic bypass LSP.

A dynamic bypass provides an automatic creation of a bypass LSP protecting any MPLS interface, based on the user configuration. Unlike static bypass LSPs, dynamic bypasses protect only one interface at a time.

The backup path of FRR LSPs give static bypasses preference over dynamic bypasses. When there are existing static bypasses which can satisfy the backup constraints, then static bypasses are used; otherwise, dynamic bypass creation is initiated when the system configuration allows.

FIGURE 20 Dynamic bypass protected interface and possible dynamic bypasses at a FRR LSP PLRs



When establishing a facility protected LSP with link or node protection, each LSR on the primary path verifies if there are any existing bypass LSPs that satisfy backup path constraints. Backup constraints include MP, backup bandwidth, hop-limit, priority, exclude links, and excluded nodes. The protected LSP backup path uses this bypass to reach its merge point. When there is no bypass available, the PLR node computes and establishes a new bypass LSP, addressing the backup path constraints.

When creating a bypass LSP for an interface, any number of facility protected LSPs may share the same bypass LSP, providing it satisfies the FRR backup constraints. Adaptive dynamic bypass LSPs can also be re-optimized periodically using the make-before-break procedure. A make-before-break switch from the current instance of a bypass to a new Instance releases all the backup paths it is attached to. All backup LSPs again perform a re-query of the backup path bypass search.

Dynamic bypass global and interface level configuration parameters, such as bandwidth, hop-limit, and priority are used while creating a new dynamic bypass LSP. Any modifications on these global and interface level configuration parameters are taken into consideration during the next cycle of re-optimization or during manually initiated bypass LSP re-optimization.

Backup path optimization does not happen while a backup is already using a bypass LSP. For the backup path to move to an optimal bypass LSP, the backup path must relinquish the current bypass LSP and go through the backup query process again. This query process searches for an optimal bypass LSP for the backup.

The bandwidth of the newly triggered bypass LSP is zero by default unless it has an explicit configuration at interface level or the backup has the backup bandwidth request. When a new facility protected LSP requests a bandwidth which cannot be accommodated within an

existing dynamic bypass LSP, there is no automatic make-before-break for the existing dynamic bypass LSP. Instead, a new dynamic bypass is created, depending on the configurations and system limits.

Similar to static bypasses, backup paths from multiple facility protected LSPs can use dynamic bypasses providing they share the same protected-interface, merge point, and satisfy FRR backup constraints.

Dynamic bypass LSP creation

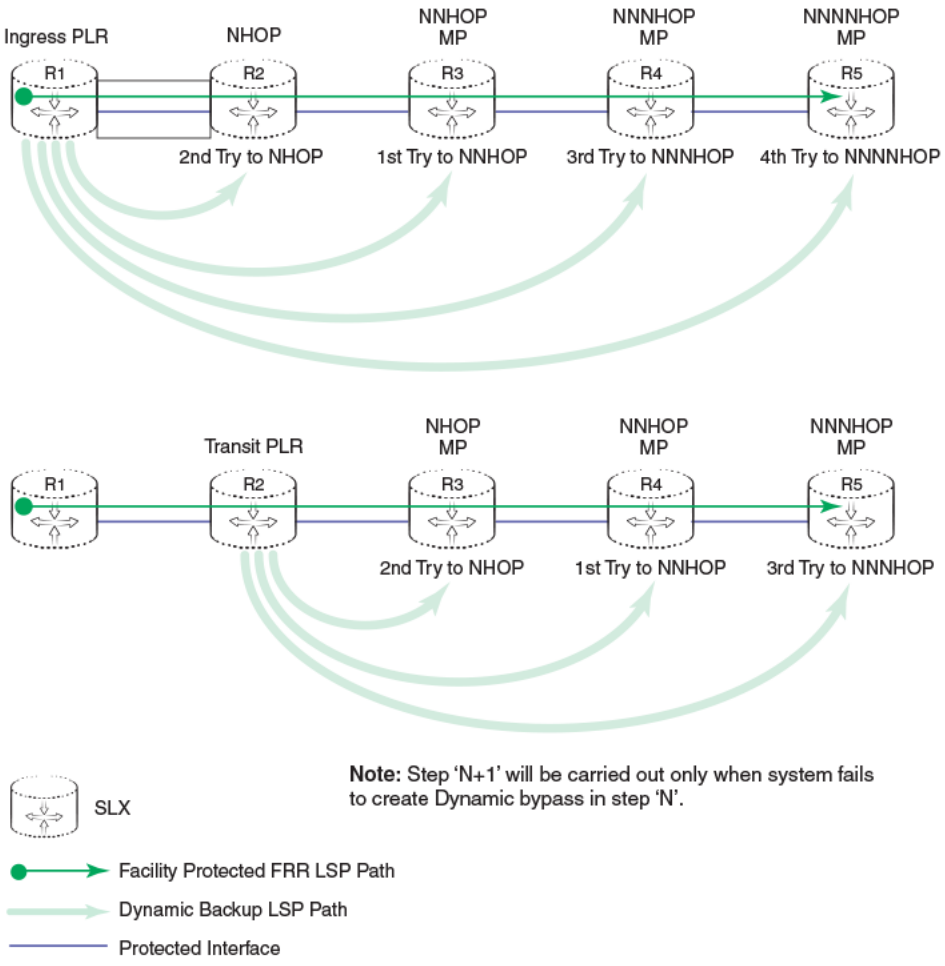
Control of dynamic bypass LSP creation is by a combination of global level configurations and interface level configurations. An MPLS interface with a dynamic bypass enabled in it is able to create a dynamic bypass LSP to provide a backup path for the protected LSP traversing out of that interface.

Creation of a dynamic bypass LSP to a destination MP follows the same order as the FRR merge point selection order.

Dynamic bypass LSP creation must meet the following criteria:

- There are no existing static bypass or dynamic bypass LSPs to satisfy the facility protected LSP backup path request.
- The creation of the dynamic bypass is allowed under the current configuration for the protected interface.
- The creation of the dynamic bypass does not exceed the configured or default system limits under the current state.
- There is an available path to setup the dynamic bypass LSP that can fulfill the backup request constraints.
- Facility protected LSP PLR originate the dynamic bypass LSP with a merge point as the destination.
- Merge point selection for dynamic bypass creation uses the existing order of the backup path merge point selection. (When not requesting node protection, reverse the order of step 1 and 2.)
    1. Tries to create a dynamic bypass LSP to the NNHOP merge point.
    2. When step 1 fails, then attempts to create a dynamic bypass LSP to the NHOP merge point.
    3. When step 2 fails, then attempts to create a dynamic bypass LSP to the NNNHOP merge point.
    4. When step 3 fails, then attempts to create a dynamic bypass LSP to the NNNNHOP merge point.
    5. This continues until the protected LSP destination node is the merge point.

FIGURE 21 Dynamic bypass creation order



The user can create a maximum of 500 bypasses in a system. This includes both static bypasses and dynamic bypasses.

Dynamic bypass naming

An automatic naming format based on type of protection it provides uniquely names the dynamic bypass LSPs. Below is the format of the dynamic bypass LSP naming scheme.

For link protection:

name prefix - protected interface local IP address - 4-digit-number

Example

```
dbyp-12.1.1.1-888
```

For node protection:

name prefix - protected interface local IP address - MP node router ID IP address - 4-digit-number

Example

```
dbyp-12.1.1.1-2.2.2.2-999
```

When looking at the name of the dynamic bypass LSP, it is possible to determine if the dynamic bypass LSP is a link protecting dynamic bypass LSP or node protecting dynamic bypass LSP. If the dynamic bypass LSP name has only one IP address in it then consider it as a link protection dynamic bypass LSP. If it has two IP addresses in its name then consider it as a node protection dynamic bypass LSP.

Additional useful information in this format is that the node protecting dynamic bypass LSP has the merge point IP address in it, so that user can easily know the MP router of dynamic bypass LSP just by looking at the name of the dynamic bypass LSP.

NOTE
Dynamic bypass LSP naming is a default behavior. The user can override the naming scheme.

Deletion of dynamic bypass LSPs

Dynamic bypass LSPs delete automatically whenever they go down. Reasons for them to go down is same as any regular LSP to go down. In addition, dynamic bypass LSPs can also delete automatically

- When not used by or associated with any backup paths for a time interval of 390 seconds.
- When a dynamic bypass is scheduled for re-optimization and there are no associated backup paths.
- The dynamic bypass is disabled at the interface level for the protected interface of the dynamic bypass LSP.
- The dynamic bypass is disabled globally in the MPLS router mode.

Dynamic bypass configurations

The configurations steps for dynamic bypass LSPs are as follows:

1. Enable the dynamic bypass on MPLS router mode.
2. **Optional step** Set the global dynamic bypass configurable parameters.
3. **Optional step** To enable a dynamic bypass on all the MPLS interfaces without going to each interface use the **enable-all-interfaces** command in the global mode. Otherwise, go to next step to enable a dynamic bypass on individual MPLS interfaces and customize the way the dynamic bypass get created for a protected interface. The user can also override the **enable-all-interfaces** commands effect on individual MPLS interfaces by configuring the dynamic bypass in those interfaces explicitly as in the next steps.
4. Enable a dynamic bypass on one or more MPLS interfaces. This step is optional when using step 3.
5. **Optional step** Set interface level dynamic bypass configurable parameters.

All modifications to the dynamic bypass interface or the router mode configuration parameters apply to the new creation of dynamic bypass LSPs.

Dynamic bypass parameter changes made at the interface level only apply to the existing dynamic bypass LSPs protecting this interface, when triggered by events such as the re-optimization timer expiry or user intervention.

The configurable parameters include, but are not limited to bandwidth, hop-limit, priority, cos, adaptiveness, and primary path. These apply to all dynamic bypass LSPs created to protect this interface.

Dynamic bypass global configuration

Globally enabling a dynamic bypass

Using the **dynamic-bypass** command in MPLS router configuration mode for the first time enables the dynamic bypass in the MPLS router. When using the **dynamic-bypass** command in the MPLS router configuration mode, which is already configured, there is no change in the existing status, enabled or disabled, of the global dynamic bypass.

To globally enable a dynamic bypass, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Globally enable the dynamic bypass.

```
device(config-router-mpls)# dynamic-bypass
```

The following example combines the steps above to globally enable dynamic bypass.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# dynamic-bypass
device(config-router-mpls-dynamic-bypass)#
```

Disabling global dynamic bypass

Use the **disable** command under router MPLS dynamic bypass mode to disable dynamic bypass in MPLS router without deleting the global mode configurations.

To disable global dynamic bypass, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Globally enable the dynamic bypass.

```
device(config-router-mpls)# dynamic-bypass
```

4. Globally disable the dynamic bypass.

```
device(config-router-mpls-dynamic-bypass)# disable
```

   This command brings down and deletes all the existing dynamic bypasses in the system.

The following example combines all of the steps above to globally disable the dynamic bypass.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# dynamic-bypass
device(config-router-mpls-dynamic-bypass)# disable
```

To enable the dynamic bypass, use the **no disable** command.

Enabling dynamic bypass on all interfaces

Use the **enable-all-interfaces** command to enable dynamic bypass on all MPLS interfaces on a router. This is applicable to all MPLS interfaces where the user has not configured dynamic bypass manually.

To enable dynamic bypass on all interfaces, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable dynamic bypass for configuration.

```
device(config-router-mpls)# dynamic-bypass
```

4. Configure dynamic bypass with the enable-all-interfaces option.

```
device(config-router-mpls-dynamic-bypass)# enable-all-interfaces
```

the following example combines all of the steps above to enable bypass on all interfaces using the **enable-all-interfaces** option.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# dynamic-bypass
device(config-router-mpls-dynamic-bypass)# enable-all-interfaces
device(config-router-mpls-dynamic-bypass)#
```

Setting the maximum number of dynamic bypass LSPs

The maximum number of dynamic bypass LSPs is configurable in the global mode. This is the limit for the total number of dynamic bypass LSPs that the system can create on a router. This number must be less than, or equal to, the global maximum number of bypass LSPs that can be configured on a router.

The device supports a maximum number of 500 bypass LSPs. It is the total number of Bypass LSPs (user created static bypass LSPs + system created Dynamic Bypass LSPs). This means that the maximum number of configurable dynamic bypass LSPs on a system is always (500 - (number of configured static bypass lsps + number of system created dynamic bypass LSPs)).

When the **max-bypasses** limit changes to a value which is less than the current active number of dynamic bypasses, the limit changes to the new value and this limit is considered for the following new creations. The existing exceeding number of dynamic bypasses do not delete.

To set the maximum number of bypasses in a MPLS interfaces, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable dynamic bypass for configuration.

```
device(config-router-mpls)# dynamic-bypass
```

4. Configure dynamic bypass using the **max-bypass** option. In this example the maximum number of bypasses is configures to *150*.

```
device(config-router-mpls-dynamic-bypass)# max-bypasses 150
```

The following example combines the steps above to set the maximum number of dynamic bypass LSPs to 150.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# dynamic-bypass
device(config-router-mpls-dynamic-bypass)# max-bypasses 150
```

Setting the maximum number of dynamic bypass LSPs per merge point

Use the **max-bypasses-per-mp** command to set the maximum number of dynamic bypass LSPs that create for this MPLS interface and reaches to any merge point. This is the limit for the total number of dynamic bypass LSPs that create to a merge point. When there is no configuration for the parameter under the interface mode, the global **max-bypasses-per-mp** parameter value is considered for this parameter.

A PLR may have 'M' number of merge points with respect to a protected LSPs. There may be 'N' number of protected LSPs riding on an interface with dynamic bypass enabled. Max-bypasses configurations limits the maximum number of dynamic bypass LSPs to each merge point.

When the **max-bypasses-per-mp** limit changes to a value which is less than the current active number of **dynamic-bypasses-per-mp**, then the limit changes to the new value and is in use for the next new creations. Existing dynamic bypasses exceeding this number do not delete.

To set the maximum number of dynamic bypass LSPs per merge point, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is *2/8* .

   ```
   device(config-router-mpls)# mpls-interface ethernet 2/8
   ```

4. Enable the interface level dynamic bypass.

   ```
   device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
   ```

5. Configure the maximum number of bypasses per merge point. In this example, the **max-bypasses-per-mp** configuration is five (*5*).

   ```
   device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# max-bypasses-per-mp 5
   ```

The following example combines the steps above to set the maximum number of dynamic bypasses per merge point to five (*5*).

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# max-bypasses-per-mp 5
```

Setting the reoptimizer-timer

When the user configures the re-optimization value to a non-zero value, in seconds, the **reoptimizer-timer** command enables the dynamic bypass LSP re-optimization.

The re-optimization timer value is configurable on all MPLS interface modes. The global set value is applicable to all dynamic bypass LSPs for which the corresponding interface level re-optimization timer value is not set.

To configure the reoptimizer-timer, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable dynamic bypass for configuration.

```
device(config-router-mpls)# dynamic-bypass
```

4. Configure the **reoptimizer-timer**. In this example, the **reoptimizer-timer** is configured to *300* seconds.

```
device(config-router-mpls-dynamic-bypass)# reoptimize-timer 300
```

The following example combines the steps above to configure the **reoptimizer-timer** to *300* seconds.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# dynamic-bypass
device(config-router-mpls-dynamic-bypass)# reoptimize-timer 300
```

Configuring adaptiveness to dynamic bypass LSPs

Dynamic Bypass LSPs are by default, adaptive in nature. To create a dynamic bypass LSP with a non-adaptive nature, use the **adaptive** command with the **enable** or **disable** option. Based on the value of the parameter, the command creates dynamic bypass LSPs as an adaptive bypass LSP or non-adaptive bypass LSP.

To enable adaptiveness to dynamic bypass LSPs, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected Ethernet interface. In this example, the selected Ethernet interface is *2/8* .

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the adaptive command with the enable option to create an adaptive bypass LSP.

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# adaptive enable
```

The following example combines the steps above to create an adaptive bypass LSP,

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# adaptive enable
```

Configuring administrative groups

Use the interface level **exclude-any** | **include-all** | **include-any** command to configure administrative groups for dynamic bypass LSPs to be created corresponding to a protected link.

Use the group-number or group-name parameter for the following:

- **Include-all** *group-number* or *group-name* : include the entire administrative group-number or group-name.
- **include-any** *group-number* or *group-name* : include the entire administrative group-number or group-name.
- **exclude-any** *group-number* or *group-name* : include the entire administrative group-number or group-name.

To configure administrative groups, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable MPLS-capable interfaces and specify the selected Ethernet interface. In this example, the selected Ethernet interface is *2/8* .

   ```
   device(config-router-mpls)# mpls-interface ethernet 2/8
   ```

4. Enable the interface level dynamic bypass.

   ```
   device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
   ```

5. Configure the administrative groups with any of the following choices. All three commands can be used simultaneously.
   - **Include-all** command. In this example, the administrative groups *4* and *5* are included in the configuration.

     ```
     device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# include-all 4 5
     ```

   - **Include-any** command. In this example, the administrative groups *6* and *7* are included in the configuration.

     ```
     device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# include-any 6 7
     ```

   - **exclude-any** command. In this example, the administrative groups *10* and *11* are included in the configuration.

     ```
     device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# exclude-any 10 11
     ```

The following example combines the steps above the configure administrative groups.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# include-all 4 5
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# include-any 6 7
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# exclude-any 10 11
```

Dynamic bypass interface configurations

Enabling dynamic bypass on a MPLS interface

Use the **dynamic-bypass** command to manually enable dynamic bypass on a MPLS interface. This enables the system to consider the MPLS interface as a protected interface and create dynamic bypasses for it. When the user configures a dynamic bypass on a MPLS interface using this command, this is a user configured interface level dynamic bypass configuration.

Dynamic bypass LSP is created to protect an interface only when

1. the dynamic bypass is globally enabled, AND
2. global dynamic bypass **enable-all-interfaces** is configured, OR the interface level dynamic bypass is enabled.

Dynamic bypass is disabled, by default, in the interface mode unless it is enabled through the global configured **enable-all-interfaces** command. There is no change in the dynamic bypass configured state (enabled or disabled) when it is already configured on the interface.

When the dynamic bypass is enabled on the interface through the global **enable-all-interfaces** command, this command changes the interface status to the user-configured interface level dynamic bypass configuration.

When the user configures the interface level dynamic bypass to the disabled status, this command retains the existing disabled state.

To enable dynamic bypass on MPLS interfaces, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected Ethernet interface. In this example, the selected Ethernet interface is *2/8*.

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)#
```

The following example combines the steps above to enable the interface level dynamic bypass for Ethernet interface *2/8*.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)#
```

Disabling dynamic bypass on MPLS interfaces

Use the **disable** command to manually disable dynamic bypass on a MPLS interface without deleting the interface level dynamic bypass configurations. This command brings down and deletes all the existing dynamic bypasses protecting the interface. Dynamic bypass configurations on the can be re-enabled by using **no** form of this command.

To disable dynamic bypass on a MPLS interface, complete the following task.

1. Enable the device for configuration.

```
device>configure
```

   Steps 1 through 4 enables dynamic bypass on a MPLS interface.

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected Ethernet interface. In this example, the selected Ethernet interface is *2/8* .

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass

```
device(config-router-mpls-if-ethernet-2/8)# dynamic bypass
```

5. Disable dynamic bypass on MPLS Ethernet interface *2/8* .

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# disable
```

The following example combines the steps above to disable dynamic bypass on a selected Ethernet interface. In this example, the selected interface is *2/8* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# disable
```

To re-enable the dynamic bypass, use the **no disable** command.

Setting the maximum number of dynamic bypasses per interface

Use the **max-bypasses** command to set the maximum number of dynamic bypass LSPs the user can create for this MPLS interface. This is the limit for the total number of dynamic bypass LSPs the user can create for this protected MPLS interface. When there is no configuration for this parameter under the interface mode, the global **max-bypasses** parameter value is considered for this parameter. This parameter value must be less than the globally set **max-bypasses** value.

To set the maximum number of dynamic bypasses per interface, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is *2/8* .

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the dynamic bypass maximum bypasses. In this example, the configuration for the maximum number of bypasses is five (*5*).

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# max-bypasses 5
```

The following example combines the steps above to set the maximum number of dynamic bypasses for Ethernet interface *2/8* to five *5* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# max-bypasses 5
```

Setting the maximum number of dynamic bypass LSPs per merge point

Use the **max-bypasses-per-mp** command to set the maximum number of dynamic bypass LSPs that create for this MPLS interface and reaches to any merge point. This is the limit for the total number of dynamic bypass LSPs that create to a merge point. When there is no configuration for the parameter under the interface mode, the global **max-bypasses-per-mp** parameter value is considered for this parameter.

A PLR may have 'M' number of merge points with respect to a protected LSPs. There may be 'N' number of protected LSPs riding on an interface with dynamic bypass enabled. Max-bypasses configurations limits the maximum number of dynamic bypass LSPs to each merge point.

When the **max-bypasses-per-mp** limit changes to a value which is less than the current active number of **dynamic-bypasses-per-mp**, then the limit changes to the new value and is in use for the next new creations. Existing dynamic bypasses exceeding this number do not delete.

To set the maximum number of dynamic bypass LSPs per merge point, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is *2/8* .

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the maximum number of bypasses per merge point. In this example, the **max-bypasses-per-mp** configuration is five (*5*).

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# max-bypasses-per-mp 5
```

The following example combines the steps above to set the maximum number of dynamic bypasses per merge point to five (*5*).

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# max-bypasses-per-mp 5
```

Specifying the name prefix

Use the **name-prefix** interface command to specify a name prefix for the dynamic bypass LSPs to be created for the MPLS protected interface. When configured, the dynamic bypass LSPs have their LSP names starting with this name prefix, appended by interface IP, merge point IP, and instance number.

The **name-prefix** configuration is allowable only when there no existing dynamic bypasses corresponding to a dynamic bypass interface. When the user wants to change the **name-prefix**, the user must disable the dynamic bypass on the interface and reconfigure the **name-prefix**, then re-enable the dynamic bypass on the interface.

To specify the **name-prefix** for a dynamic bypass LSP, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected Ethernet interface. In this example, the selected Ethernet interface is 2/8.

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic bypass
```

5. Configure the **name-prefix** command. In this example, the **name-prefix** configuration is *mydps* . This is the default name for the prefix string.

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# name-prefix mydps
```

The following example combines the steps above to specify the **name-prefix** for a dynamic bypass LSP. In this example, the selected interface is *2/8* , and the **name-prefix** is *mydps* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# name-prefix mydps
```

Setting the priority

Use the **priority** command to configure the setup and holding priority for the dynamic bypass LSPs to be created for the MPLS protected interface. Priority levels can be from zero to seven for a dynamic bypass LSP corresponding to a protected link. By default, setup priority is seven, and the hold priority is zero. When the interface mode priority values are not configured, and there are riding backups on the dynamic bypass, the dynamic bypass re-optimization new holding priority is the maximum priority of the currently riding backups.

To set the priority, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is *2/8* .

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the priority command. In this example, the first number, 6, is the set-up priority and the second number, 3, is the holding-priority.

```
device(config-router-mpls-if-ethernet-2/8—dynamic-bypass)# priority 3 6
```

The following example combines the steps above to configure the setup and holding priority for the dynamic bypass LSPs to be created for the MPLS protected interface. The **setup-priority** is configured to *6* , and the **holding-priority** is configured to *3* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8—dynamic-bypass)# priority 6 3
```

Enabling and disabling the record route option

An interface level **record** route parameter can be configured for a dynamic bypass LSP corresponding to a protected link. Use the **record enable** or **disable** command to enable or disable the dynamic bypass LSP record route options. Based on the value of this parameter, dynamic bypass LSPs create with their record route option enabled or disabled.

To enable or disable the **record** route option, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected Ethernet interface. In this example, the selected Ethernet interface is *2/8* .

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the **record** route command.
   - To enable the **record** route command.

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# record enable
```

   - To disable the **record** route command.

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# record disable
```

The following example combines the steps above to enable or disable the **record** route command.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# record enable
-or-
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# record disable
```

Setting traffic engineering bandwidth parameters

Traffic engineering rates define the bandwidth parameters for the dynamic bypass LSP to be created corresponding to a protected link. The dynamic bypass LSP bandwidth is based on this command's mean-rate and protected LSP backup path requested bandwidth.

When there is no configuration for the interface mode mean-rate value, then all dynamic bypass LSPs create with the bandwidth the same as the backup path requested bandwidth. This means that system tries to create a dynamic bypass LSP with backup path requested bandwidth and dynamic bypass LSP bandwidths varying from one to the other, based on the riding backup.

When the interface mode configured mean-rate value is 0 kbps, such as when it is explicitly configured as 0 by use, then the system can create dynamic bypasses for backup bandwidth requests of 0 kbps only. When the backup path bandwidth is more than zero, then such a request does not lead to the creation of a new dynamic bypass LSP. This option provides a way for the user to limit the dynamic bypass creations to only non-bandwidth protected backups.

When the interface mode configuration mean-rate value is a non-zero value, then the system does not create dynamic bypasses for the backups which request backup bandwidth that is more than the interface mode configured value. When the backup bandwidth is less than, or equal to, the configured value, then such request can be honored to ride an existing dynamic bypass or create a new dynamic bypass. With this configuration, all the newly created dynamic bypasses have a fixed bandwidth (value same as interface mode user configured non-zero mean-rate value).

A mean-rate value more than current interface reservable bandwidth is not recommended. Configuration succeeds with the new value even when it is more than the interface reservable bandwidth.

Setting tie-break option for dynamic bypass CSPF

A user can create an interface level tie-breaking option for the CSPF calculation of dynamic bypass LSPs to be created for the protected MPLS interface. Use this option for the dynamic bypass LSP path computation tie breaking procedure.

To create an interface level tie-breaking option for the CSPF calculation of a dynamic bypass LSP, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is 2/8.

   ```
   device(config-router-mpls)# mpls-interface ethernet 2/8
   ```

4. Enable dynamic bypass

   ```
   device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
   ```

5. Configure the **tie-breaking** command with the **least-fill** option.

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# tie-breaking  least-fill
```

The following example combines the steps above to configure the **tie-breaking** command to use the **least-fill** option for the CSPF calculation of dynamic bypass LSPs.

```
device>configure
device(config)# router mpls
device(cnfig-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# tie-breaking  least-fill
```

Setting the hop limit for dynamic bypass CSPF

The user can configure an interface level hop-limit for dynamic bypass LSPs corresponding to a protected link.

The user can compute a dynamic bypass path, so the hop-limit is the minimum of the backup requested hop limit and the interface mode configured hop limit. This computed hop limit is set as the dynamic bypass LSP hop limit during the initial creation of the dynamic bypass.

At the time of re-optimization, the dynamic bypass hop limit is modified so it is at its minimal ((hop limits of all riding backup paths), interface configuration hop-limit).

To set the hop limit for dynamic bypass CSPF, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is *2/8*.

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the hop limit. In this example, the hop limit is configured to four (*4*).

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# hop-limit 4
```

The following example combines the steps above to set the hop limit for dynamic bypass CSPF to four (*4*).

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# hop-limit 4
```

Setting CoS for dynamic bypass LSPs

The user can configure an interface level CoS with a value of 0-7 for when creating a dynamic bypass LSP for the protected MPLS interface. Use this CoS value to create dynamic bypasses corresponding to the interface.

To set CoS for dynamic bypass LSPs, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is *2/8*.

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the CoS value. In this example, the CoS value is configured to five (*5*).

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# cos 5
```

The following example combines the steps above to set the CoS value for dynamic bypass LSPs to five (*5*).

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# cos 5
```

Setting the from-address for dynamic bypass LSPs

The user can configure an interface level **from** IP address when creating a dynamic bypass LSP for the protected MPLS interface . Use the dynamic bypasses **from** address as the IP address.

To set the **from** address for dynamic bypass LSPs, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is *2/8*.

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the **from** address. In this example, the from address configuration is *11.11.11.11*.

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# from 11.11.11.11
```

The following example combines the steps above to configure the **from** command IP address for dynamic bypass LSPs to *11.11.11.11*.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# from 11.11.11.11
```

Setting the explicit path for dynamic bypass LSPs

The user can configure an explicit path for the dynamic bypass LSPs for a protected interface. Use the **primary-path** command for this purpose.

To set the explicit path for dynamic LSPs, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable MPLS-capable interfaces and specify the selected Ethernet interface. In this example, the selected Ethernet interface is *2/8*.

   ```
   device(config-router-mpls)# mpls-interface ethernet 2/8
   ```

4. Enable the interface level dynamic bypass.

   ```
   device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
   ```

5. Configure the **primary-path** command. In this example, the primary path name selected is *dbyp-path*.

   ```
   device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# primary-path dbyp-path
   ```

The following example combines the steps above to set an explicit path with the name of *dbyp-path*, for dynamic bypass LSPs.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# primary-path dbyp-path
```

Setting CSPF computation mode for dynamic bypass LSPs

The user can configure the CSPF computation mode for the CSPF path calculation of dynamic bypass LSPs for a protected interface using the **cspf-computation-mode** command. CSPF can make use either the TE metric as cost or the IGP metric as cost for shortest path first algorithm. The TE metric is the default.

To set the CSPF computation mode, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable MPLS-capable interfaces and specify the selected interface. In this example, the selected Ethernet interface is *2/8*.

```
device(config-router-mpls)# mpls-interface ethernet 2/8
```

4. Enable the interface level dynamic bypass.

```
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
```

5. Configure the **cspf-computation-mode** command. In this example, the **use-igp-metic** option is selected.

```
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# cspf-computation-mode use-igp-metric
```

The following example combines the steps above to set the CSPF computation mode for dynamic bypass LSPs to the **use-igp-metric** option.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 2/8
device(config-router-mpls-if-ethernet-2/8)# dynamic-bypass
device(config-router-mpls-if-ethernet-2/8-dynamic-bypass)# cspf-computation-mode use-igp-metric
```

Bypass LSP RSVP IGP synchronization

RSVP IGP synchronization applies to bypass LSPs. Upon IGP (OSPF/ISIS) neighborship down, a bypass LSP can either go down or creates a new instance per RSVP IGP synchronization. A new instance creation does not happen when there are active backups on the bypass LSP. All other behaviors are normal LSP behaviors regarding RSVP IGP synchronization.

The user can enable RSVP IGP synchronization by using the **handle-ospf-neighbor-down** or the **handle-isis-neighbor-down** command in MPLS router policy configuration mode.

To configure bypass LSP RSVP IGP synchronization, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable the policy mode.

```
device(config-router-mpls)# policy
```

4. Configure the RSVP IGP synchronization. In this example, use the **handle-ospf-neighbor-down** command.

```
device(config-router-mpls-policy)# handle-ospf-neighbor-down
```

The following example combines the steps above for bypass LSP RSVP IGP synchronization using the **handle-ospf-neighbor-down** command.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# handle-ospf-neighbor-down
```

Bypass LSP statistics

Bypass tunnel statistics measure the amount of traffic pushed into the bypass tunnel. It accounts the traffic for both tunnels and transit which are in a repaired state and using the bypass tunnel to forward traffic.

The user can enable bypass LSP statistics with the **ingress-tunnel-accounting** command.

To enable bypass LSP statistics, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enter the policy mode.

```
device(config-router-mpls)# policy
```

4. Enable bypass LSP statistics with the **ingress-tunnel-accounting** command.

```
device(config-router-mpls-policy)# ingress-tunnel-accounting
```

The following example combines the steps above to enable bypass LSP statistics using the **ingress-tunnel-accounting** command.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# ingress-tunnel-accounting
```

The user can display the statistics for bypass LSPs by using the **show mpls statistics tunnel** show command.

```
device# show mpls statistics tunnel
Tunnel                          Total           Since Last clear   Rate
Index  LSP Name      Proto  Byp  Packets Bytes   Packets Bytes      Pkts/sec  Bytes/sec
     1 my-bypass-lsp    R   Yes        0     0         0     0             0          0
     2 t2               R   No         0     0         0     0             0          0
```

Link protection for FRR

To avoid loss of traffic, Fast Reroute (FRR) protects the LSP and allows a broken LSP to be repaired immediately at the point of failure.

A Label Switched Path (LSP) set up across a MPLS network is used to switch traffic across MPLS network. The path used by an LSP across the network is based upon network resources or any other traffic engineering constraints provided by the user. Based on TE-constraints, the ingress MPLS router computes the path to be taken by LSP and signals it using RSVP protocol.

By nature, nodes and links in a MPLS network are prone to failure. It is likely that the link or the nodes through which LSP is traversing can fail. In the event of a failure of a node or link, RSVP protocol has mechanisms that inform the ingress node about the failure the to ingress node. On receipt of failure message for LSP across the path, the ingress router re-signals the LSP using a new path.

Due to messaging and other network delays, the ingress router cannot respond fast enough to minimalize the loss of traffic. Traffic is lost from the moment the failure occurs and until the new path is setup for the LSP, which is quite large in quantum for service provider networks.

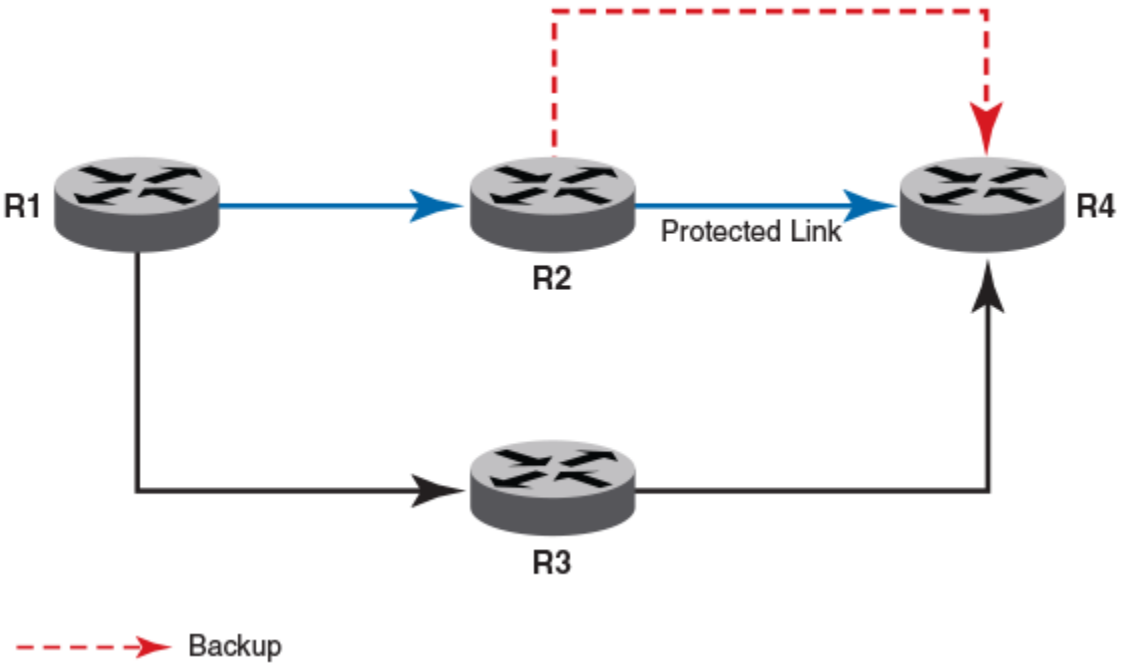
To avoid loss of traffic, Fast Reroute (FRR) protects the LSP and allows a broken LSP to be repaired immediately at the point of failure. The point of failure is termed as "Point of local repair" (PLR), where the LSP can be repaired locally without intimating or waiting for the

ingress router. PLR is the MPLS router which detects the failure and redirects the traffic appropriately to its backup path with minimal loss.

Typically at the PLR, two type of protection can be provided to LSP:

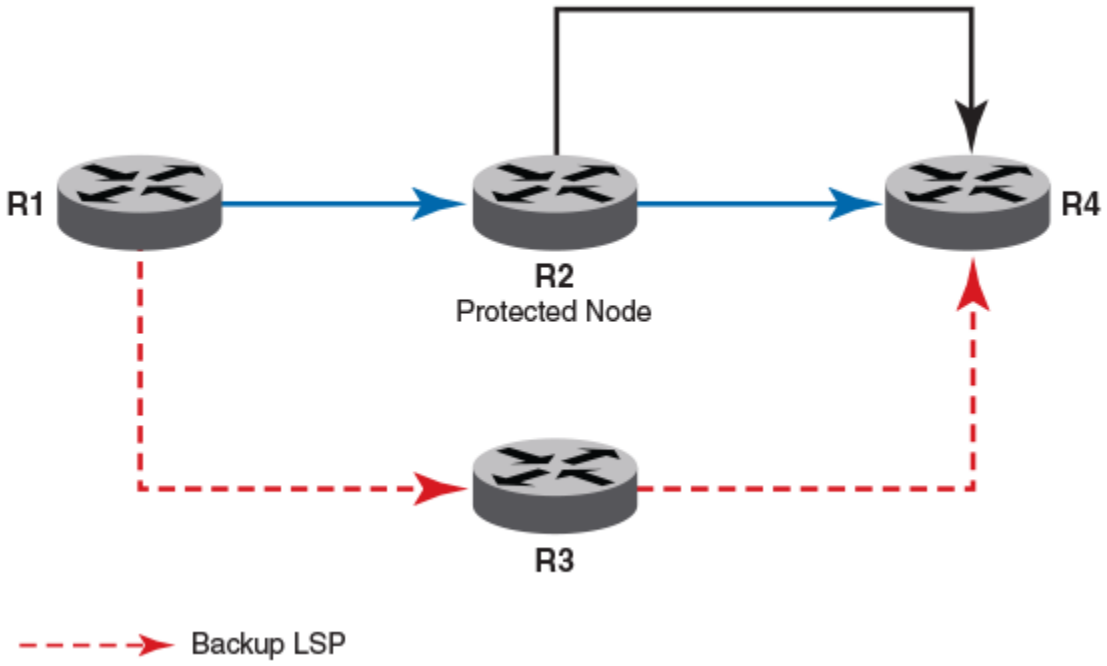
Link Protection: In this protection, the backup is selected in such a way that it avoids the failed link which was used earlier by the LSP. Traffic merges back to the main stream from the backup on the very next MPLS router. Refer to following Link protection for FRR illustrating link protection provided at R2 to LSP ingressing from R1 to R4.

FIGURE 22 Link protection



Node Protection: In this protection, backup is selected in such a way that it avoids the failed link along with router to which this link connects. The node which was responsible for link failure is avoided altogether in its entirety, which was used earlier by the LSP. Traffic merges back to main stream from backup on somewhere downstream from the node, which is being avoided. Refer to Link protection for FRR illustrating node protection provided at R1 to LSP ingressing from R1 to R4.

FIGURE 23 Node protection



As part of link protection for FRR, ingress routers are allowed to expose this property of MPLS RSVP LSP to the user and lets the user choose between link protection or node protection. Once the node protection is chosen, PLR first tries to establish a backup LSP, which provides node protection. When node protection is not possible, it attempts to fall back to link protection.

When the user chooses link protection over node protection, this is communicated to all routers participating in LSP. Each PLR. in this case. limits its search for backup LSP, which provides link protection. In cases where link protection cannot be offered, PLR falls back to node protection.

Link protection for FRR applies to both one to one protection and many to one FRR protection.

Link protection for FRR provides options to the user to set a preferential method requested for local protection. When RSVP LSP is enabled with FRR (local protection), the user would be able to configure either link protection or node protection. Node protection remains the default.

Configuration steps for adaptive and non-adaptive LSPs have inherent differences on their make-before-break capabilities. The default behavior for both types of LSPs remains node protection.

Configuring protection type preference for non-adaptive LSPs

The user can change the protection type preference (node protection to link protection or link protection to node protection) only in an administratively down state of a non-adaptive LSPs. Any non-adaptive LSP, which is already enabled by the user for signaling, cannot be changed.

Configuring protection type preference for Adaptive LSPs

Because adaptive LSPs TE-property can be changed without restarting LSP and changed values takes effect through the make-before-break process, you are allowed to change the protection type preference (node protection to link protection or link protection to node

protection) at any point of time during life cycle of adaptive LSPs, irrespective of its administrative or operational state. When you change the preferential protection type and it commits to the configuration, configuration takes effect. Signaling of the changed property depends on the state of LSP. For example, when the administrator is UP or DOWN, it is operationally UP or DOWN. There is no change in the MBB trigger because of link protection for FRR. All MBB aspects including, but not limited to, implicit and explicit commits remain unchanged.

TABLE 6 Protection types for adaptive LSPs

Requesting Node Protection			Requesting Link Protection	
	Earlier Request: Link protection.	Earlier Request: Node protection.	Earlier Request: Link protection.	Earlier Request: Node protection.
Adaptive LSP	LSP requests node protection on next commit operation.	No change	No change	LSP requests link protection on next commit operation.
Non-Adaptive disabled LSP	LSP requests node protection once the user enables LSP.	No change	No change	LSP requests link protection once the user enables LSP.

NOTE

If the user tries to configure link protection for FRR on a non-adaptive enabled LSP, the following error is displayed:

```
Error: Must disable lsp before changing parameters
```

Configuring an adaptive LSP

The Fusion software supports adaptive LSPs. When configuring an adaptive LSP, the user can change the following parameters of an LSP while it is in the enabled state:

- CSPF
- exclude-any
- hop-limit
- include-all
- include-any
- primary-path
- priority
- tie-breaking
- traffic-eng

When one of these parameters is changed on a Adaptive LSP, a new instance of the same LSP is signaled using the newly defined parameters. Once the new LSP comes up, traffic is moved to the new LSP instance and the old LSP instance is torn down.

To configure an LSP named *to20* as an adaptive LSP, complete the following steps.

1. Configure the device

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the signaled label switched path (LSP).

```
device(config-router-mpls)# lsp to20
```

   In this example, the LSP name is *to20* .

4. Enable the LSP to be modified without exiting.

```
device(config-router-mpls-lsp-to20)# adaptive
```

The following example configures an LSP named *to20* as an adaptive LSP.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp to20
device(config-router-mpls-lsp-to20)# adaptive
```

Once an LSP is configured to be adaptive, it can have the parameters described above changed. In the following example, the setup and hold priorities for adaptive **lsp** *to20* are changed to seven and one.

```
device(config-router-mpls)# lsp to20
device(config-router-mpls-lsp-to20)# priority 7 1
```

The new parameters are not changed for the adaptive LSP until the **commit** command is issued for the LSP.

> NOTE
> Once the **commit** command has been issued, there may be a 30 millisecond traffic disruption.

In the following example of the **show mpls lsp** command for **lsp** *to20* , the priorities are not changed in the output.

```
device(config-router-mpls-lsp-to212)# show mpls lsp to212
LSP to212, to 10.5.1.1
 From: 10.4.1.1, admin: UP, status: UP, tunnel interface: tnl1
  Times primary LSP goes up since enabled: 1
  Metric: 0, number of installed aliases: 0 Adaptive
  Maximum retries: 0, no. of retries: 0
  Setup priority: 7, hold priority: 0
  Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
  Constraint-based routing enabled: yes
  Tie breaking: random, hop limit: 0
  OTHER INSTANCE PRIMARY: NEW_INSTANCE admin: DOWN, status: DOWN
  Maximum retries: 0, no. of retries: 0
  Setup priority: 7, hold priority: 1
  Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
  Constraint-based routing enabled: yes
  Tie breaking: random, hop limit: 0
  Active Path attributes:
  Tunnel interface: tnl1, outbound interface: e1/2
  Tunnel index: 4, Tunnel instance: 1 outbound label: 3
  Path calculated using constraint-based routing: yes
  Explicit path hop count: 1
    10.2.1.2 (S)
  Recorded routes:
   Protection codes: P: Local N: Node B: Bandwidth I: InUse
   10.2.1.2
```

The following **commit** command makes the new parameter settings active in the adaptive **lsp** *to20* configuration.

```
device(config-router-mpls)# lsp to20
device(config-router-mpls-lsp-to20)# commit
```

After the **commit** command runs, the user can see that the priorities have changed by using the **show mpls lsp** command for **lsp** *to20* .

```
device(config-router-mpls-lsp-to212)# show mpls lsp to212
LSP to212, to 10.5.1.1
 From: 10.4.1.1, admin: UP, status: UP, tunnel interface: tnl1
 Times primary LSP goes up since enabled: 1
 Metric: 0, number of installed aliases: 0 Adaptive
 Maximum retries: 0, no. of retries: 0
 Setup priority: 7, hold priority: 1
 Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
 Constraint-based routing enabled: yes
 Tie breaking: random, hop limit: 0
 Active Path attributes:
  Tunnel interface: tnl1, outbound interface: e1/2
  Tunnel index: 4, Tunnel instance: 2 outbound label: 3
  Path calculated using constraint-based routing: yes
```

```
Explicit path hop count: 1
 10.2.1.2 (S)
Recorded routes:
 Protection codes: P: Local N: Node B: Bandwidth I: InUse
 10.2.1.2
```

Re-optimizing LSPs

Under ordinary conditions, an LSP path does not change unless the path becomes inoperable. Consequently, the router needs to be directed to consider configuration changes made to an LSP and to optimize the LSP path based on those changes. This is accomplished using the **mpls re-optimize** command as shown in the following.

```
device# mpls reoptimize lsp to20
```

NOTE

On re-optimization of an adaptive LSP, LSP accounting statistics might miss the accounting of some of the packets.

Time-triggered re-optimizing

The user can set a timer to optimize a specific LSP path on a periodic basis.

Upon expiration of this timer, the LSP is optimized for a new path when the new path has a lower cost than the existing path. This timer can be configured when the LSP is in a disabled state, and the timer value can be adaptively changed when the LSP is in an enabled state by issuing a **commit** to take effect. Until a **commit** is issued the re-opt timer is disabled.
To set the LSP re-optimization timer, complete the following steps.

1. Configure the device.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Configure the signaled label switched path (LSP).

   ```
   device(config-router-mpls)# lsp to20
   ```

   In this example, the selected LSP name is *to20* .

4. Configure the re-optimize timer.

   ```
   device(config-router-mpls-lsp-to20)# reoptimize-timer 1000
   ```

   in this example, the re-optimize time is configured to 1000 seconds which specifies the number of seconds from the beginning of one re-optimization attempt to the beginning of the next attempt.

In the following example, the LSP re-optimize timer is configured to 1000 seconds.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp to20
device(config-router-mpls-lsp-to20)# reoptimize-timer 1000
```

Configuring a re-optimization timer does not interfere with running the manual **reoptimize** command.

NOTE
When upgrading software, the configured adaptive LSPs are initialized with the no re-optimization timer.

NOTE
Time-triggered re-optimizing does not apply to LSPs within a FRR network.

RSVP LSP with FRR

RSVP LSP with fast reroute (FRR) protection for detour backup.

Detour backup establishes backup path with detour sessions from point of local repair (PLR) to merge point (MP).

RSVP FRR LSP with detour backup applies the same algorithm for the backup path computation. The algorithm is designed to follow RFCs to avoid various issues intrinsic to detour backup type such as early session merging and bandwidth sharing.

NOTE
Fusion devices support FRR failover for RSVP LSPs. If the point of failure is a transit router with respect to the LSP path, then failover can be performed (LSP is repaired, and traffic flow restored) very quickly, in less than 50 milliseconds. When the point of failure is ingress router with respect to the LSP path, FRR failover is not quick enough and it may take more than 50 milliseconds to repair the LSP. Traffic loss will be more than 50 milliseconds.

RSVP per-session statistics

Resource Reservation Protocol (RSVP) statistics are a useful troubleshooting tool. At the global and interface levels, statistics are already available. In some cases, especially scaled scenarios, it is helpful to also count these statistics on a per-session level. The same statistics that are available at the global and interface levels are also available at the session level.

RSVP per-session statistics and their applicability

The table below illustrates the collection of RSVP protocol statistics to the session available at the global, interface, and session levels.

TABLE 7 RSVP statistics and their applicability

Statistics Name	Global	Interface	Session
Protocol Packet (stats-Path, Resv, PathErr, ResvErr, PathTear, ResvTear, pathconf)	Yes	Yes	Yes
Path Expired	Yes	No	Yes
Resv Expired	Yes	No	Yes
Unknown Message Type	Yes	Yes	No

TABLE 7 RSVP statistics and their applicability (continued)

Statistics Name	Global	Interface	Session
Bundle, Ack, Hello, Sumrefresh Packets	Yes	Yes	No
Bad Authorization	Yes	Yes	No
Message Id Error	Yes	Yes	No
Summary Refresh Error	Yes	Yes	No
Nack	Yes	Yes	No
Errored Protocol Packets	Yes	No	Yes
Errored length	Not Applicable	Not Applicable	Not Applicable
Errored version	Not Applicable	Not Applicable	Not Applicable
Errored chksum	Not Applicable	Not Applicable	Not Applicable
Errored malloc	Not Applicable	Not Applicable	Not Applicable

RSVP-TE Hello

The RSVP-TE Hello feature is an optional extension to RSVP-TE protocols to detect neighbor down scenarios. It makes use of Hello messages as a keepalive poll mechanism between RSVP peers on a link.

A failure along the path of a signaled RSVP-TE LSP can remain undetected for two minutes or longer (reservation or RESV time-out). During this time, bandwidth is held by the non-functioning LSP on the nodes downstream from the point of failure along the path with the intact state. If this bandwidth is needed by head-end tunnels to signal or re-signal LSPs, tunnels may fail to come up for several minutes, thereby negatively affecting convergence time.

Hello messages enable RSVP nodes to detect when a neighboring node is not reachable. When the RSVP-TE Hello protocol notices that a neighbor is not responding, it treats it as a neighbor down case (link layer communication failure) and deletes the LSP state or reroutes it, based on the type of LSP. This action frees the node's resources so they can be reused by other LSPs.

A Hello message is sent out periodically to each RSVP peer on a link. If no response is received from the peer within a specified period, then the peer is announced "dead" (down). RSVP LSPs going over that peer must either be torn down or re-routed, based on the nature of the LSPs.

This Hello mechanism is intended for use between immediate neighbors. Hello processing between two neighbors supports independent selection of configurations of failure detections intervals.

The configuration of Hello message is completely optional. All the messages may be ignored by nodes which do not wish to participate in Hello message processing.

By default, this feature is disabled.

RSVP-TE Hello extension composition

The Hello extension is composed of three parts:

- Hello Message
- Hello REQUEST object
- Hello ACK object

Each neighbor can individually issue Hello REQUEST objects. Each request may be answered by an Hello ACK object. The Hello extension is designed so that one side can use the mechanism while the other side does not. All messages may be ignored by nodes

which do not wish to participate in Hello message processing. If a particular peer never responds to Hello messages, Extreme routers do not assume that the peer is dead, but simply assume that it does not support Hello messages.

The Hello message has a Msg Type of 20 with a message format as follows:

```
Hello Message : := Common Header [ INTEGRITY ]
Hello
```

RSVP-TE Hello process

When both sides of a link support wish to participate in Hello message processing, the RSVP-TE Hello process follows this procedure.

1. A node periodically generates a Hello message containing a HELLO REQUEST object for each neighbor whose status is being tracked. The hello-interval governs this periodicity. There is support for each interface configuration of RSVP-TE HELLO to be flexible. This value may be configured on a per-interface basis. The default value is nine seconds, and the configurable range of hello-interval is 1 to 60 seconds.
2. When generating a message containing a HELLO REQUEST object, the sender fills in the **Src_Instance** field with a value representing its per neighbor instance. This value does not change while the agent is exchanging Hellos with the corresponding neighbor. The sender also fills in the **Dst_Instance** field with the *Src_Instance* value most recently received from the neighbor. For reference, refer to this variable as the *Neighbor_Src_Instance*. If no value has ever been received from the neighbor or this node considers communication to the neighbor to have been lost, the *Neighbor_Src_Instance* is set to zero (0). The generation of a message must be suppressed when a HELLO REQUEST object is received from the destination node within the prior hello-interval interval.
3. On receipt of a message containing a HELLO REQUEST object, the receiver generates a Hello message containing a HELLO ACK object. The receiver also verifies that the neighbor has not reset. This is done by comparing the sender's **Src_Instance** field value with the previously received value. If the *Neighbor_Src_Instance*value is zero, and the **Src_Instance** field is non-zero, the *Neighbor_Src_Instance* is updated with the new value. If the value differs, then the node treats the neighbor as if communication has been lost.
4. The receiver of a HELLO REQUEST object also verifies that the neighbor is reflecting back the receiver's Instance value. This is done by comparing the received **Dst_Instance** field with the **Src_Instance** field value most recently transmitted to that neighbor. If the neighbor continues to advertise a wrong non-zero value after a configured number of intervals (hello-tolerance), then the node must treat the neighbor as if communication has been lost.
5. On receipt of a message containing a HELLO ACK object, the receiver must verify that the neighbor has not reset. This is done by comparing the sender's **Src_Instance** field value with the previously received value. If the *Neighbor_Src_Instance*value is zero, and the **Src_Instance** field is non-zero, the *Neighbor_Src_Instance* is updated with the new value. If the value differs or the **Src_Instance** field is zero, then the node must treat the neighbor as if communication has been lost.
6. The receiver of a HELLO ACK object must also verify that the neighbor is reflecting back the receiver's Instance value. If the neighbor advertises a wrong value in the **Dst_Instance** field, then a node must treat the neighbor as if communication has been lost.
7. If no Instance values are received, through either REQUEST or ACK objects, from a neighbor within a configured number of hello-intervals (hello-tolerance), then a node must presume that it cannot communicate with the neighbor. The default for this number is three (3). So, the time-out is equal to three times the retransmission period. The range for hello-tolerance is 1 to 255.
8. When communication is lost or presumed to be lost, a node may re-initiate HELLOs. If a node does re-initiate, t must use a *Src_Instance*value different than the one advertised in the previous HELLO message. This new value must continue to be advertised to the corresponding neighbor until a reset or reboot occurs, or until another communication failure is detected. If a new instance value has not been received from the neighbor, then the node must advertise zero in the **Dst_Instance**value field.

For those sessions going over the interface on which a neighbor down is detected, the following actions are taken by the nature of the LSP:

- For RSVP sessions with no backup available, these sessions are brought down.
- For RSVP sessions with available backups, FRR switchover is performed.

The HELLO mechanism is intended for use between immediate neighbors. So, when the HELLO messages are being exchanged between immediate neighbors, the IP TTL field of all outgoing HELLO messages is set to 1.

RSVP-TE Hello considerations

Configuring hello-interval on both ends of a link

The **hello-interval** command at a mpls-interface level is used to configure the interval time for sending RSVP-TE Hello request messages. Configuring the Hello-interval allows the interface to initiate Hello request messages. When both ends of the link are configured to respond to RSVP-TE Hello messages, the neighbor receiving the request message generates an ACK message.

Configuring hello-interval only on one end of a link

The **hello-interval** command at a mpls-interface level is used to configure the interval time for sending RSVP-TE Hello request messages. If the neighbor does not wish to participate in RSVP-TE Hello message communication, it can ignore the Hello request messages. The neighbor may send out the ACKs only if it chooses to participate in the RSVP-TE Hello messages. If a particular peer never responds to Hello messages, do not assume that the peer is dead, but simply assume that it does not support Hello messages.

Removing Hello support from one end of the link

Consider the case when both ends of the link supported RSVP-TE Hello messages, and the exchange of messages was normal as both links were up. Remove the support for Hello from one side of the link. The other side keeps sending Hello Request messages, but the neighbor starts ignoring these requests as it no longer wishes to participate in Hello messages exchange. In this case, because the neighbor stops sending ACKs, the router considers this as a neighbor down case and brings down all the RSVP sessions going over that interface. After a neighbor down event, Hello message exchange starts off from scratch (re-initiates). If the neighbor does not respond to Hello requests, the router assumes that the neighbor does not support Hello because no ACK was ever received after re-initiating Hello.

Also, when RSVP Hello is supported only on one end of the link, the end that supports Hello sends Hello request messages until it hits the *hello_tolerance* limit, then stops sending any further Hellos messages. It restarts sending Hellos when it receives a Hello message from the neighbor and then again continues the two-way communication as before.

NOTE

Caution: When disabling RSVP Hello, disable it on both sides of the link at the same time to avoid bringing down all the RSVP sessions going over that link.

Configuring Hello-tolerance

Hello-tolerance can be individually configured on both ends of the interface. Considering both sides of the link are participating in Hello communication, if no Instance values are received, through either of the REQUEST or ACK objects, from a neighbor within this configured hello-tolerance number of hello-intervals, then this node presumes that it cannot communicate with the neighbor.

Configuring hello-acknowledgments

Configuring **hello-acknowledgments** command (on the global MPLS RSVP Hello level) enables the router to respond back by sending Hello ACKs on neighbors not carrying any RSVP sessions. By default, HelloACKs are sent only to neighbors carrying RSVP sessions.

Creating an LSP

To create a signaled LSP and enter the LSP configuration level, complete the following steps.

1. Enable the device to be configured.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

In this example, the enabled LSP is named *tunnel1*.

```
device(config-router-mpls-lsp)#
```

The following example shows how to create a signaled LSP. In this example the signaled LSP name is *tunnel1*.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp)#
```

Specifying the egress LER

The egress LER is the router from which packets exit the MPLS domain in this LSP.

Each LSP requires one and only one egress LER. After the LSP is successfully established, the address of the egress LER is installed as an internal host route on the ingress LER, allowing the ingress LER to direct BGP next-hop traffic into the LSP. The destination address does not necessarily have to be the final node in the primary path specified for the LSP. When the final node in the path differs from the destination address, the hop between the final node in the path and the egress LER is treated as a loose hop.

To specify *10.100.1.1* as the address of the egress LER for LSP *tunnel1*, complete the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable the specified label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

In this example, the specified LSP is *tunnel1*.

4. Specify node as the address of the egress LER for a LSP.

```
device(config-router-mpls-lsp)# to 10.100.1.1
```

   In this example, address *10.100.1.1* is configured as the address of the egress LER for LSP *tunnel1*.

The following example shows how to specify *10.100.1.1* as the address of the egress LER for LSP *tunnel1*.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp)# to 10.100.1.1
```

NOTE

When IS-IS is used as the IGP, the egress LER advertises the tunnel destination must be in Extended IP Reachability TLV 135 in order for the LSP to be properly mapped by CSPF. To ensure that this happens, connect to the egress LER and enable IS-IS on the interface which has the IP address of the tunnel destination. When none of the interfaces on the egress LER has the IP address of the tunnel destination (for example, when the tunnel destination address is the egress LER's router ID) rather than an interface address -- to manually set the router ID, then the tunnel destination address must be included in Traffic Engineering router ID TLV 134 in the LSP originated by the egress LER. This is accomplished by setting the egress LER's traffic engineering policy to IS-IS with the **traffic engineering isis level** command

Specifying a source address for an LSP

The user can optionally specify a source IP address for a signaled LSP. RSVP path messages carry this address.

To specify a source IP address of 10.2.3.4 for LSP tunnel1, complete the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Assign the signaled label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   In this example, the specified LSP is *tunnel1*.

4. Specify the source IP address.

```
device(config-router-mpls-lsp-tunnel1)# from 10.2.3.4
```

   In this example, the specified source address is *10.2.3.4*.

The following examples show IP address *10.2.3.4* being assigned as the source IP address for LSP *tunnel1*.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# from 10.2.3.4
```

The **from** command specifies the source IP address to be carried in RSVP Path messages for the LSP. This command is optional. When the from command is specified, then the address is always carried in RSVP path messages as the source IP address for the LSP. When the **from** command is not specified, and when the LSP is enabled, the device dynamically determines the source IP address of the LSP (using the device's first loopback as the source IP address).

> NOTE
> A loopback interface must be configured for RSVP to use it as source IP address when the from command is not specified.

The IP address specified in the **from** command affects only the IP address carried in the RSVP path messages for the LSP. It does not affect the outgoing interface (and thus the actual path) that the path messages are sent out.

Configuring redundant paths for an LSP

A signaled LSP has a primary path, which is either user-defined or computed by the ingress LER.

> NOTE
> This section describes the behavior of redundant paths. However, the user can exercise further control over the path selection process by specifying the path selection mode and preferred path using the **select-path** command.

Optionally, the user can configure one or more redundant paths to serve as a backup. When the primary path fails, traffic for the LSP can be forwarded over the redundant path. When no redundant path is configured for the LSP, and when the primary path fails, the ingress LER automatically attempts to compute a new path to the egress LER, establish the new path, and then redirect traffic from the failed path to the new path.

Configuring a redundant path allows the user to exercise greater control over the rerouting process than when the ingress LER simply calculated a new path to the egress LER. When a redundant path is configured, when the primary path fails, the ingress LER attempts to establish the redundant path. As with the primary path, a redundant path follows an explicit route of loose or strict hops.

By default, the redundant path is established only when the primary path fails. The user can optionally configure a redundant path to operate in hot-standby mode. A hot-standby path is established at the same time the primary path in the LSP is established. Resources are allocated to the hot-standby path, although no packets for the LSP are sent over the hot-standby path until the primary path fails. When the primary path fails, the already-established hot-standby path immediately takes over from the primary path. Since the hot-standby path is already active, service outages that can arise from the process of signaling and establishing a new path are eliminated.

After the redundant path has been activated, the ingress LER continues to try to connect to the egress LER over the primary path, either indefinitely or up to the configured retry limit. When a connection over the primary path can be established, the redundant path is deactivated, and traffic for the LSP is again sent over the primary path. Once the primary LSP becomes available again, the redundant path is torn down; when the path is a hot-standby path, it reverts to its backup status.

The user can configure multiple redundant paths. When the primary path fails, the ingress LER attempts to establish a connection to the egress LER using the first redundant path configured for the LSP. When a connection cannot be established using the first redundant path, the second redundant path is tried, and so on. When a connection cannot be established after trying each redundant path in the configuration, the first redundant path is tried again, and the process repeats. This behavior can be further modified using the **select-path** command.

To configure a secondary path, first create a path. After the user creates the path, the user can specify that it is to be used as a redundant path. For example, complete the following steps to cause a path called *alt_sf_to_sj* to be used when the primary path in LSP *tunnel1* fails.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable and configure signaled label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   In this example, the LSP specifies is LSP *tunnel1*.

4. Enable and configure secondary path.

```
device(config-router-mpls-lsp-tunnel1)# secondary-path alt_sf_to_sj
```

   In this example, the secondary path's name is *alt_sf_to_sj* .

5. Enable hot-standby mode.

```
device(config-router-mpls-lsp-sec-path)#standby
```

   Issuing the **secondary-path** command enters the secondary path configuration level. From this level, the user can specify that this path is to operate in hot standby mode.

The following example enables a path called *alt_sf_to_sj* to be used when the primary path in LSP *tunnel1* fails. Once the LSP is enabled, both the primary and hot-standby paths are activated, although packets are directed over only the primary path.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# secondary-path alt_sf_to_sj
device(config-router-mpls-lsp-sec-path)# standby
```

NOTE

At the secondary path level, the user can configure separate values for the following parameters: Class of Service (CoS), setup and hold priority, bandwidth allocations, and inclusion or exclusion of interfaces in administrative groups. When the user does not configure these parameters at the secondary path level, the secondary path uses the default values for these parameters.

Configuring path selection

The user can exercise control over the paths used by an LSP by setting the select mode and by specifying a preferred path using the select-path command.

By default, an LSP with primary and secondary paths configured immediately uses the primary path. When the primary path fails, a secondary (redundant) path is used. When the primary path comes back up, traffic reverts to the primary path and the secondary (redundant) path returns to a back-up state. However, path selection can be configured to operate in any of the following three modes:

- **auto select mode –** This is the default mode of the router and no special configuration is required. When this mode is operating, the router always tries to use the primary path to carry traffic when the primary path has stayed operating in the working state for at least the amount of time specified in revert-timer configuration command. When **no revert-timer** is configured for the LSP, a value of zero seconds is used which causes immediate switching of the path.

- **manual select mode –** In this mode, traffic is switched to a user-specified path after the selected path has stayed operating in the working state for at least the amount of time specified in the **revert-timer** configuration. In the **manual** select mode, traffic stays on the selected path as long as the path remains in working condition and only switches to an alternative path, such as the primary path when the selected path experiences a failure. Once the selected path comes back into working condition for the amount of time specified by the **revert-timer** configuration, traffic is switched back.

When an LSP is configured in **manual** select path mode with at least one other hot standby secondary path, the operation is as follows: when the selected path goes down, the system tries to bring up one hot standby secondary path to protect the primary path, but when the selected path is up, system brings down the hot standby secondary path since the selected path is already serving as a hot standby for the primary path.

unconditional select mode – In this mode, traffic is switched to and stays on the selected path regardless of the path's condition even when it is in a failure state. The main difference between **manual** and **unconditional** select mode is the test of the working condition of the user selected path. When configured in **unconditional** mode, the router starts the signaling for the selected path if has not already done so and brings down all other paths; this includes the primary path and the path carrying traffic when it is not the selected path. Because the speed at which the selected path comes up cannot be guaranteed, traffic forwarding might be disrupted.
The **auto** select mode and **manual** select mode configurations use the **revert-timer** configuration.

The following steps configure the LSP named *samplelsp* with a primary path named *pathprimary* and two secondary paths named *pathsecondarya* and *pathsecondaryb* . The path named *pathsecondaryb* is configured as a selected path in the **manual** select mode.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable and configure the signaled label switched path (LSP).

```
device(config-router-mpls)# lsp samplelsp
```

   In this example, the LSP is called *samplelsp* .

4. Configure the primary path.

```
device(config-router-mpls-lsp-samplelsp)# primary-path pathprimary
```

   In this example, the primary path selected is called *pathprimary* .

5. Configure the first secondary path.

```
device(config-router-mpls-lsp-samplelsp)# secondary-path pathsecondarya
```

   In this example the first primary path configured is called *pathsecondarya*.

6. Configure the second secondary path.

```
device(config-router-mpls-lsp-samplelsp)# secondary-path pathsecondaryb
```

   In this example, the secondary path configured is called *pathsecondaryb*.

7. Configure the selected path in the **manual** mode.

```
device(config-router-mpls-lsp-samplelsp)# select-path manual pathsecondaryb
```

8. Apply the parameter modifications to the LSP.

```
device(config-router-mpls-lsp-samplelsp)# commit
```

After configuring this example, traffic for *samplelsp* travels over the *pathsecondaryb* path whenever this path is in working condition because the **no revert-timer** has been configured. When a **revert-timer** is configured, the router waits for the *pathsecondaryb* path to be up for at least the amount of time specified in the configuration of the **revert-timer** command. When the select mode is changed to **unconditional**, as shown below, traffic is switched to the pathsecondaryb path, regardless of its working condition.

9. Change the select mode to **unconditional** for path pathsecondaryb.

```
device(config-router-mpls-lsp-samplelsp)# select-path unconditional pathsecondaryb
```

The following example configures the LSP named *samplelsp* with a primary path named *pathprimary* and two secondary paths named *pathsecondarya* and *pathsecondaryb* . The path named *pathsecondaryb* is configured as a selected path in the **manual** select mode. After the parameters modification are made to the LSP, the select mode is changed to **unconditional**, and the traffic is switched to the *pathsecondaryb* path.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp samplelsp
device(config-router-mpls-lsp-samplelsp)# primary-path pathprimary
device(config-router-mpls-lsp-samplelsp)# secondary-path pathsecondarya
device(config-router-mpls-lsp-samplelsp)# secondary-path pathsecondaryb
device(config-router-mpls-lsp-samplelsp)# select-path manual pathsecondaryb
device(config-router-mpls-lsp-samplelsp)# commit
device(config-router-mpls-lsp-samplelsp)# select-path unconditional pathsecondaryb
```

Configuration changes made to the select mode do not take effect for an already enabled LSP until the change is activated implicitly using the **commit** command, explicitly using a **reoptimize** command, or a system reboot is performed.

NOTE

When the user configures a primary path to be the selected path, a message is generated that states that it is already the default system behavior because the primary path is the default preferred path. In this instance, no configuration information is saved in the configuration file.

Configuring a path selection revert timer

The path selection revert timer provides an option to stabilize a path before traffic is switched to it.

Without a configured path selection revert timer, the router switches between a primary and secondary path immediately after the current working path goes down. A problem with this mode of operation is that it can cause flapping when the current path goes up and down frequently. Also, the LSP to which the route is switching traffic might be unstable, which causes the router to fail back to the current LSP almost immediately.

The path revert timer insures the stability of the LSP to which the traffic is switched by specifying the number of seconds that the LSP must be running before it carries traffic.

To configure a path selection revert timer for an LSP, complete the following steps using the **revert-time** command.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable and specify signaled label switched path (LSP).

```
device(config-router-mpls)# lsp samplelsp
```

In this example, *samplelsp* is the selected LSP.

4. Configure revert timer.

```
device(config-router-mpls-lsp-samplelsp)# revert-timer 10
```

The *timer-value* value is the number of seconds that the router waits after the primary, or selected path comes up before traffic reverts to that path. in this example, the value is configured to 10 seconds.

The following example configures the path selection revert time to 10 seconds for LSP named 'samplelsp'.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# lsp samplelsp
device(config-router-mpls-lsp-samplelsp)# revert-timer 10
```

When deploying the **revert-time** command, consider the following:

- The **revert-time** command has no effect on the unconditional select mode. Traffic is unconditionally switched to the user-selected path and stays on it.
- The path stability test used with the revert timer is based on the uptime of the latest instance of the path. This value can be different when the selected path has gone through a "make-before-break" procedure.
- For an LSP going through re-optimization, the new LSP does not carry traffic until the revert timer expires.
- When a user changes the revert time, the basis of counting is the uptime of the path and is independent of the sequence or combination of configurations. Take, for example, a path that is configured in the manual select mode to be a secondary path with a revert-time of 10 seconds. After the secondary path comes up, a 10-second timer starts, but after five seconds, the user changes the revert-timer value to four. Now the path has already been stable beyond the newly configured revert time, so the original time is canceled, and traffic immediately switches over. However, if the user were to change the revert-time value to eight seconds after running for five seconds, the existing count would terminate and start a new count of three seconds from the moment the first count terminated.

Usage considerations:

- The **revert-time** command has no effect on the unconditional select mode. Traffic is unconditionally switched to the user-selected path and stays on it.
- The path stability test used with the Revert Timer feature is based on the uptime of the latest instance of the path. This value can be different when the selected path has gone through a "make-before-break" procedure.
- For an LSP going through re-optimization, the new LSP does not carry traffic until the revert timer expires.
- When a user changes the revert timer, the basis of counting is the uptime of the path and is independent of the sequence or combination of configurations. Take, for example, a path that is configured in the manual select mode to be a secondary path with a revert-timer of 10 seconds. After the secondary path comes up, a 10-second timer starts, but after five seconds, the user changes the revert-timer value to four. Now the path has already been stable beyond the new configured revert-timer, so the original timer is canceled and traffic immediately switches over. However, if the user were to change the revert-timer value to eight seconds after running for five seconds, the existing count would terminate and start a new count of three seconds from the moment the first count terminated.

Specifying the primary path for an LSP

The primary path is the route that packets generally travel when going through an LSP.

The user can specify a user-defined path or no path at all. Once the LSP is enabled, the ingress LER attempts to signal the other LSRs in the path so that resources can be allocated to the LSP. When the user does not specify a primary path, the path used in the LSP is the shortest path to the egress LER, as determined from standard IP routing methods, or CSPF when it is enabled.
In the following example, to specify the *sf_to_sj* path as the primary path for LSP *tunnel1*, complete the following steps.

1. Enable the device for configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable and configure the specified label switched path (LSP).

   ```
   device(config-router-mpls)# lsp tunnel1
   ```

   In this example, the specified LSP is *tunnel1*.

4. Configure the primary path.

   ```
   device(config-router-mpls-lsp-tunnel1)# primary-path sf_to_sj
   ```

   In this example, the primary path is defined as *sf_to_sj*.

The following example configures the primary path as *sf_to_sj* for LSP *tunnel1*.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# primary-path sf_to_sj
```

Configuring signaled LSP parameters

Once the user has configured a path, the user can configure signaled LSPs that see it. An LSPs configuration can specify not only the path that label-switched packets follow in a network, but also the characteristics of the path, the resources allocated along the path, and actions applied to the packets by the ingress or egress LERs.

The user can perform the following tasks when configuring a signaled LSP:

- Performing a Commit for an LSP
- Creating the LSP
- Specifying an egress LER for the LSP
- Specifying a primary path for the LSP (optional)
- Configuring secondary or hot-standby paths for the LSP (optional)
- Setting aliases for the egress LER (optional)
- Setting a Class of Service (CoS) value for the LSP (optional)
- Allocating bandwidth to the LSP (optional)
- Configuring the setup and hold priority for the LSP (optional)
- Setting a metric for the LSP (optional)

- Including or excluding administrative groups from LSP calculations (optional)
- Limiting the number of hops the LSP can traverse (optional)
- Specifying a tie-breaker for selecting CSPF equal-cost paths (optional)
- Disabling the Record-Route function (optional)
- Disabling CSPF path calculations (optional)
- Configure Maximum Packet Size without fragmentation
- Enabling the LSP
- Disabling the LSP
- Generating Traps and Syslogs for LSPs

Performing a commit for an LSP configuration command

For LSP configuration commands to take effect, either an explicit or implicit commit must be performed. These are performed as shown in the following:

Performing an explicit commit

The user can perform an explicit commit within the configuration of a specified LSP using the **commit** command. The following example demonstrates the creation of an LSP named *samplelsp* and its primary and secondary paths. After the configuration is entered, the commit command is executed to activate the configuration.

Complete the following steps to create an LSP and its primary and secondary paths.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable signaled label switched path (LSP).

```
device(config-router-mpls)# lsp samplelsp
```

   In the example, the enabled LSP is named *samplelsp*.

4. Configure primary path.

```
device(config-router-mpls-lsp-samplelsp)# primary-path pathprimary
```

   In this example, the primary path is named 'pathprimary'.

5. Configure a secondary path.

```
device(config-router-mpls-lsp-samplelsp)# secondary-path pathsecondarya
```

   In this example, a secondary path is named *pathsecondarya*.

6. Configure another secondary path.

```
device(config-router-mpls-lsp-samplelsp)# secondary-path pathsecondaryb
```

   In this example, a second secondary path is configured and named 'pathsecondaryb'.

7. Configure a manual path selection mode.

```
device(config-router-mpls-lsp-samplelsp)# select manual pathsecondaryb
```

   In this example, the manual path selection is secondary path named 'pathsecondaryb".

8. Activate the configuration.

```
device(config-router-mpls-lsp-samplelsp)# commit
```

The following example shows an explicit commit within the configuration of a specified LSP (*samplelsp*) using the **commit** command.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp samplelsp
device(config-router-mpls-lsp-samplelsp)# primary-path pathprimary
device(config-router-mpls-lsp-samplelsp)# secondary-path pathsecondarya
device(config-router-mpls-lsp-samplelsp)# select manual pathsecondaryb
device(config-router-mpls-lsp-samplelsp)# select manual pathsecondaryb
device(config-router-mpls-lsp-samplelsp)# commit
```

The **reoptimize** command is another type of explicit commit.

Using the **reoptimize** command, the user can activate all pending LSP configuration changes for specified LSP or use the **all** option to activate all pending LSP configuration changes for all of the LSPs configured on the router.

Performing an implicit commit

MPLS allows the user to modify the configurable parameters for RSVP LSPs while the LSP is operational.

After modifying the parameters for an operational LSP, the user must execute the **commit** command to apply the changes. Applying these configuration changes requires a new instance of the LSP to be signaled with a modified or new set of parameters, also known as make-before-break. Once the new instance of the LSP is up, the old instance is removed.

By default, if the adaptive parameters of an LSP have changed, but the changes are not yet committed, any system-initiated make-before-break, such as an LSP re-optimization event, is ignored. To allow changes to be automatically applied, the user can use the **implicit-commit lsp-reoptimize-timer** command under the router MPLS policy command to enable certain types of events to trigger implicit commit.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable policy parameter configuration.

```
device(config-router-mpls)# policy
```

4. Enable implicit commit.

```
device(config-router-mpls-policy)# implicit-commit lsp-reoptimize-timer
```

   The following example enable the LSP re-optimize timer to trigger an implicit commit.

```
device# configure
device(confoig)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# implicit-commit lsp-reoptimize-timer
```

Setting a Class of Service value for the LSP

The 3-bit 'EXP' field in the MPLS header can be used to define a Class of Service (CoS) value for packets traveling through the LSP. The user can manually set a CoS value for the LSP. The CoS value that the user sets is applied to the CoS (EXP) field in the MPLS header of all packets entering this LSP. This lets all packets traveling through an LSP to be treated with the same priority as they travel the MPLS domain. The user can assign the LSP a CoS in the range 0-7.
To assign a CoS value of 7 (highest priority) to all packets traveling through LSP tunnel 1, complete the following steps.

1. Enable the device for configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable and configure the signaled label switched path (LSP).

   ```
   device(config-router-mpls)# lsp tunnel1
   ```

   In this example, the specified LSP is named 'tunnel1'.

4. Assign a value for the CoS.

   ```
   device(config-router-mpls-lsp-tunnel1)# cos 7
   ```

   The MPLS CoS value is used for determining priority within an MPLS domain only, so when the label is popped, the CoS value in the MPLS header is discarded; it is not copied back to the IP ToS field.

The following example shows how to configure the **cos** command with a value of 7.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# cos 7
```

Allocating bandwidth to an LSP

Allocating bandwidth to an LSP lets the LSRs determine how much bandwidth the LSP can consume and how much of the available bandwidth resources can be advertised.

The user can specify the allocation of bandwidth for an LSP, including the maximum and average rates for packets that travel over it.

The user can specify an average mean-rate kbps for the data on the LSP. When necessary, data can travel at max-rate kbps, as long as the burst sent at the maximum rate contains no more than max-burst bytes.

To set the maximum rate of packets that can go through an LSP (in Kbps) complete the following steps.

1. Enable the device for configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable and configure the label switched path (LSP)

   ```
   device(config-router-mpls)# lsp tunnel1
   ```

   In this example, the LSP is configured to specify the LSP named 'tunnel1'.

4. Enable traffic engineering parameter configuration and configure the max-rate.

```
device(config-router-mpls-lsp-tunnel1)#)# traffic-engineering mean-rate 400 max-rate 500 max-burst
70000
```

   In this example, the mean-rate is configured to 400, the max-rate to 400, and the max-burst to 70000.

The following example shows setting the maximum rate of packets that go through the LSP in Kbps.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# traffic-engineering mean-rate 400 max-rate 500 max-burst 70000
```

To set the average rate of packets that can go through an LSP (in Kbps).

```
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# traffic-eng mean-rate 400
```

To set the maximum size (in bytes) of the largest burst the LSP can send at the maximum rate.

```
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# traffic-eng max-burst 70000
```

Configuring a priority for a signaled LSP

The priority determines the relative importance of the LSP during setup or preemption.

The user can specify a priority for each signaled LSP for which this is the ingress LER. The priority for an LSP has two components the setup priority and the hold priority.

When multiple LSPs are enabled at the same time, such as when the device is booted, LSPs that have a higher setup priority are enabled before LSPs that have a lower setup priority.

When an LSP is assigned a high setup priority, it may preempt an LSP that is already established, causing resources assigned to the lower priority LSP to be diverted to the higher priority LSP. The hold priority specifies how likely an established LSP is to give up its resources to another LSP. To be preempted, an LSP must have a lower hold priority than the preempting LSPs setup priority. In addition, an established LSP can be preempted by a higher priority LSP only if it would allow the higher priority LSP to be established successfully.

To configure LSP 'tunnel1' with a setup priority of 6 and hold priority of 1, complete the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable and configure the label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   In this example, the LSP selected is named 'tunnel1'.

4. Configure setup and hold priorities.

```
device(config-router-mpls-lsp-tunnel1)# priority 6 1
```

   In this example, the setup priority is configured to 6 and the hold priority is configured at 1. The LSP setup priority must be lower than or equal to the hold priority.

In the following example, the configures LSP 'tunnel1' with a setup priority of 6 and hold priority of 1.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# priority 6 1
```

Assigning a metric to the LSP

The user can assign a metric to the LSP, which can be used by routing protocols to determine the relative preference among several LSPs towards a given destination.

To assign a metric of five to LSP 'tunnel1', complete the following steps.

1. Enable the device for configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable and configure the label switched path (LSP).

   ```
   device(config-rouoter-mpls)# lsp tunnel1
   ```

   In this example, the selected LSP is named *tunnel1*.

4. Assign a metric of five to LSP *tunnel1*.

   ```
   device(config-router-mpls-lsp-tunnel1)# metric 5
   ```

the following example shows assigning a metric value of 5 to LSP *tunnel1*.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# metric 5
```

By default, all LSPs have a metric of one. A lower metric is preferred over a higher one. When multiple LSPs have the same destination LSR, and they have the same metric, the traffic load is shared among them.

Including or excluding administrative groups from LSP calculations

Administrative groups, also known as resource classes or link colors, lets the user assign MPLS enabled interfaces to various classes.

When a device uses CSPF to calculate the path for an LSP, it takes into account the administrative group to which an interface belongs; the user can specify which administrative groups the device can include or exclude for this calculation.

For example, to include interfaces in either of the administrative groups "gold" and "silver" in the path calculations for LSP *tunnel1*, complete the following steps.

1. Enable the device for configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable and configure the label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   in this example, the LSP selected is named *tunnel1*.

4. Configure to include selected administrative groups

```
device(config-router-mpls-lsp-tunnel1)# include-any gold silver
```

   In this example, the device includes any of the interfaces that are members of groups *gold* or *silver* when calculating the path for this LSP. Only those interfaces in the *gold* or *silver* groups are considered for the LSP.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# include-any gold silver
```

To exclude interfaces in either administrative group *gold* or *silver* when the path for LSP *tunnel1* is calculated.

In this example, the device excludes any interface that is a member of group *gold* or *silver* when it calculates the path for this LSP. Only interfaces that are not part of either group are considered for the LSP.

```
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# exclude-any gold silver
```

To specify that an interface must be a member of both the *gold* or *silver* administrative groups in order to be included in the path calculations for LSP *tunnel1*.

In this example, an interface must be a member of all the groups specified in the **include-all** command in order to be considered for the LSP. Any interface that is not a member of all the groups is eliminated from consideration.

```
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# include-all gold silver
```

Limiting the number of hops the LSP can traverse

By default, the path calculated by CSPF can consist of no more than 255 hops, including the ingress and egress LERs. The user can optionally change this maximum to a lower number.

To limit CSPF to choosing a path consisting of no more than 20 hops for LSP tunnel1, complete the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable and configure the label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   In this example, the selected LSP is named *tunnel1* .

4. Configure the limit number of hop the LSP can traverse.

```
device(config-router-mpls-lsp-tunnel1)# hop-limit 20
```

   In this example the hop-limit is set to 20.

the following example limits the CSPF to choosing a path of no more than 20 hops for LSP *tunnel1*.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# hop-limit 20
```

Specifying a tie-breaker for selecting CSPF equal-cost paths

Constrained Shortest Path First (CSPF) may calculate multiple, equal-cost paths to the egress label Edge Router (LER). When this happens, the device chooses the path whose final node is the physical address of the destination interface. When more than one path fits this description, by default, the device chooses the path with the fewest hops. When multiple paths have this number of hops, the device chooses one of these paths at random. The user can optionally configure the device to choose the path that has either the highest available bandwidth or the lowest available bandwidth.

For example, complete the following steps for CSPF to select the path with the highest available bandwidth when choosing among equal-cost paths calculated for LSP *tunnel1*.

1. Enable the device for configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Enable and configure the label switched path (LSP).

   ```
   device(config-router-mpls)# lsp tunnel1
   ```

   In this example, the LSP is configured to the LSP named *tunnel1*.

4. Select the tie breaking mode for the CSPF.

   ```
   device(config-router-mpls-lsp-tunnel1)# tie-breaking least-fill
   ```

   In this example, the path is selected on least-fill criteria.

   The *least-fill* parameter causes CSPF to choose the path with the highest available bandwidth (that is, the path with the least utilized links).

The following example causes CSPF to select the path with the highest available bandwidth when choosing among equal-cost paths calculated for LSP *tunnel1*.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# tie-breaking least-fill
```

Disabling CSPF path calculations

By default, Constrained Shortest Path First (CSPF) is enabled for signaled LSP calculations. When the device is the ingress LER for the LSP, it uses the information in the TED to help determine a path for the LSP.

To disable the constraint-based path selection for LSP *tunnel1*, complete the following steps.

1. Enable the device for configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable and configure the label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   In this example, the selected LSP is named *tunnel1*.

4. Disable CSPF.

```
device(config-router-mpls-lsp-tunnel1)# no cspf
```

The following example disables the constraint-based path selection for LSP *tunnel1*.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# no cspf
```

Disabling the record route function

The RSVP RECORD_ROUTE object (RRO) allows an LSPs path to be recorded.

An RRO consists of a series of sub-objects that can contain the addresses of the LSRs in the path. This information can be viewed with the show mpls lsp detail command. The path information is recorded in the RRO by default, but the user can disable path recording. To disable path recording in the RRO, complete the following steps.

1. enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable and configure the label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   In this example, the selected LSP is named *tunnel1*.

4. Disable the recording path routes.

```
device(config-router-mpls-lsp-tunnel1)# no record
```

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# no record
```

Configuring the maximum packet size

Configuring the maximum packet size allows the user to set a maximum IP packet size for packets that traverse an LSP without being fragmented. It can be configured for both primary and secondary paths.

To configure a maximum IP packet size for an LSP, complete the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. enable and configure the label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   In this example, the LSP specified name is *tunnel1*.

4. Configure the IP packet maximum transmission unit size.

```
device(config-router-mpls-lsp-tunnel1)# ipmtu 1500
```

   In this example, the *packet-size* variable specifies the maximum packet size in bytes for IP packets transiting the LSP without being fragmented.

The following example configures a maximum IP packet size of 1500 bytes for LSP *tunnel1*.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# ipmtu 1500
```

Enabling a signaled LSP

Enabling the LSP causes the path to be set up and resources reserved on the LSRs in the LSPs primary path.

After the user sets the parameters for the signaled LSP, the user can enable it. Enabling the LSP is the final step in configuring it.

To enable LSP *tunnel1*, complete the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the label switched path (LSP).

```
device(config-router-mpls)# lsp tunnel1
```

   In this example, the selected LSP name is *tunnel1*.

4. Establish the LSP.

```
device(config-router-mpls-lsp-tunnel1)# enable
```

The following example shows enabling LSP *tunnel1*.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# enable
```

Disabling a signaled LSP

Disabling a signaled LSP de-activates it, but does not remove the LSP from the device's configuration.

To remove the LSP from the device's configuration, use the no lsp name command. To make changes to an active LSP, first, disable the LSP, modify parameters on the LSP, and then enable the LSP.
To disable a signaled LSP *tunnel1*, complete the following steps.

1. Configure the device.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Configure the label switched path (LSP).

   ```
   device(config-router-mpls)# lsp tunnel1
   ```

   in this example the selected LSP name is *tunnel1*.

4. Tear down the LSP (disable).

   ```
   device(config-router-mpls-lsp-tunnel1)# disable
   ```

   In this example, LSP *tunnel1*, is torn down and disabled.

The following example disables LSP *tunnel1*.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp tunnel1
device(config-router-mpls-lsp-tunnel1)# disable
```

Configuring the RSVP refresh interval

To maintain path states and resource reservations on the routers in an LSP, RSVP Path and Resv messages are sent at regular intervals.

Path messages flow downstream in an LSP, from the ingress LER towards the egress LER. Resv messages flow upstream, in the reverse direction of Path messages.

The user can control how often the Path and Resv messages are sent by setting the refresh interval. By default, the refresh interval is 30 seconds. The user can set the refresh interval from 0 through 2147483 seconds.

Complete the following steps to set the refresh interval.

1. Configure the device.

   ```
   device# configure
   ```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable configuration of RSVP parameters.

```
device(config-router-mpls)# rsvp
```

4. Configure average interval between refresh path and reservation messages.

```
device(config-router-mpls-rsvp)# refresh-interval 20
```

   In this example, the refresh interval is configured at 20 seconds.

The following example shows setting the refresh interval to 20 seconds.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# rsvp
device(config-router-mpls-rsvp)# refresh-interval 20
```

Configuring the RSVP refresh multiple

The refresh multiple is the number of refresh intervals that must elapse without a refresh message before a path state or resource reservation times out.

When refresh messages are not received, RSVP path states and resource reservations are removed from the routers in an LSP. By default, the device waits the length of three refresh intervals; when no refresh message is received by the end of that time, the path state or resource reservation is removed.

The user can set the refresh multiple from zero through 65535 intervals.
Complete the following steps to set the refresh multiple to five intervals.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable configuration of RSVP parameters

```
device(config-router-mpls)# rsvp
```

4. Configure the refresh multiple.

```
device(config-router-mpls-rsvp)# refresh-multiple 5
```

   The refresh multiple is the number of unresponsive paths or reservations before time runs out. In this example, the refresh multiple is configured to 5.

The following example shows setting the refresh multiple to 5.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# rsvp
device(config-router-mpls-rsvp)# refresh-multiple 5
```

Displaying MPLS

Displaying the Traffic Engineering database

An LSRs *Traffic Engineering Database (TED)* contains topology information about nodes in an MPLS domain and the links that connect them. This topology information is obtained from the IS-IS LSPs with traffic engineering extensions. IS-IS LSPs with TE extensions have special extensions that contain information about an MPLS-enabled interface's traffic engineering metric, bandwidth reservations, and administrative group memberships.

An LSR, when configured to do so, floods IS-IS LSPs with TE extensions for its MPLS-enabled interfaces to its neighboring routers in the IS-IS area. Other LSRs store the information from the IS-IS LSPs with TE extensions in their own *Traffic Engineering Databases (TED)*, allowing each LSR in the area to maintain an identical TED describing the MPLS topology. The topology information in the TED is used by the CSPF process when it calculates traffic-engineered paths for signaled LSPs.

Displaying a traffic engineering path to a destination

The user can display a traffic engineering path to a IPv4 destination address using a specified set of resource parameters. This allows the user to gain insight into a traffic engineering path in a network, before setting it up using RSVP. This helps the user to avoid a RSVP path setup failure due to unavailable requested resources along the path to the destination host.

T o display the traffic engineering path to a IPv4 destination address, enter the following command.

```
device# show mpls te path 10.4.4.4
```

The following example displays an output from the show mpls te path command.

```
device# show mpls te path 10.12.12.12 hop-limit 2
Path to 10.12.12.12 found! Time taken to compute: 0 msec
Hop-count: 2 Cost: 2000 ISIS Level-1
     Hop 1: 10.1.0.1, Rtr 10.13.13.13
     Hop 2: 10.1.0.2, Rtr 10.12.12.12
```

Displaying signaled LSP status information

The user can display status information about signaled LSPs for which the device is the ingress LER as shown in the example below.

```
device# show mpls lsp
*: The LSP is taking a Secondary Path
Admin Oper    Tunnel    Up/Dn Retry Active
Name    To          State State Intf     Times No.    Path
t1      10.3.3.3    UP       UP*      tnl1     1        5      v2
```

NOTE

The **show mpls lsp brief** command displays the same information as the **show mpls lsp** command.

The **show mpls lsp detail** command displays detailed information about a specific LSP. To display detailed information about the status of the LSPs for which the device is the ingress LER, enter the **show mpls lsp detail** command as shown in the example below.

```
device# show mpls lsp detail
LSP t1, to 10.3.3.3
 Path selected: pathsecondaryb, mode: manual revert-timer:
 Path selected is up for 3 seconds for the latest instance, traffic will be switched to it in 7 seconds.
 From: 10.2.3.4, admin: UP, status: UP, tunnel interface: tnl1
 Times primary LSP goes up since enabled: 1
 Metric: 1, number of installed aliases: 0
 Maximum retries: 0, no. of retries: 3
 Pri. path: dir, active: no
  Setup priority: 7, hold priority: 0, ipmtu 1400
```

```
 Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
 Constraint-based routing enabled: yes
 Tie breaking: random, hop limit: 0
Sec. path: v2, active: active
 Hot-standby: no, status: up
 Setup priority: 7, hold priority: 0
 Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
 Constraint-based routing enabled: yes hop limit: 0
Active Path attributes:
 Tunnel interface: tnl1, outbound interface: e1/1
 Tunnel index: 5, outbound label: 1966
 Path calculated using constraint-based routing: no
 Explicit path hop count: 1
  10.10.10.2 (S) -> 10.20.20.2 (S) Recorded routes:
 Protection codes: P: Local    N: Node    B: Bandwidth    I: InUse
  10.10.10.2 -> 10.20.20.2
```

Displaying path information

A path is a list of router hops that specifies a route across an MPLS domain. The user can create a path and then configure the LSPs that see the path. When the LSP is enabled, the ingress LER attempts to signal the other LSRs in the path, so that resources can be allocated to the LSP.

The user can display information about the paths configured on the device as shown in the following example.

```
device# show    mpls    path
Path Name         Address           Strict/loose      Usage Count
to110_120         10.110.110.2      Strict            1
                  10.120.120.3      Strict
to2_pri           10.10.10.2        Strict            0
to2_sec           10.110.110.2      Strict            0
to3               10.110.110.2      Loose             1
                  10.120.120.3      Loose
to3_pri           10.10.10.2        Strict            1
                  10.20.20.3        Strict
to3_sec           10.110.110.2      Strict            0
                  10.120.120.3      Strict
to4               10.110.110.2      Loose             1
                  10.120.120.3      Loose
                  10.130.130.4      Loose
to_23             10.110.110.2      Strict            1
                  10.20.20.3        Strict
```

Displaying the MPLS routing table

The MPLS routing table is used to store routes to egress LERs.

To display the contents of the MPLS routing table, enter the **show mpls route** command. The port field displays whether an interface/port is either Ethernet or POS.

Displaying the MPLS forwarding information

The **show mpls forwarding** command displays MPLS forwarding information. The 'out-intf' field in the output of the **show mpls forwarding** command displays whether an interface/port is either an Ethernet port or a POS port.

Displaying MPLS configuration information

The **show runing router mpls** command displays all of the user-configured MPLS parameters. Using the **show runing router mpls** command, the user can display all of the following global parameters configured on an Extreme device:

- brief

- cspf-group
- interface
- lsp
- path

Displaying in the detail mode

The user can display all of the MPLS global information and all of the MPLS configuration information using the **show run router mpls** command. The **show run router mpls** command displays all of the detailed information.

```
device# show run router mpls
router mpls
 policy
  admin-group m2 2
  traffic-eng isis level-1
  retry-limit 22 rsvp
  refresh-interval 40
 rsvp
  refresh-interval 40

 ldp
  hello-timeout 12 ka-interval 18
  advertise-fec  list1
  session 10.30.30.6 key 1 $!dZ@

 mpls interfaces
  mpls-interface e1/1
  ldp-enable

 mpls-interface e1/2
 ldp-enable
 reservable-bandwidth percentage 80
 admin-group 2

mpls paths
path mu1_to_mu3
 strict 10.1.1.1
 strict 10.1.1.2
 strict 10.3.3.1
 strict 10.3.3.2
path mu1_to_mu2_2
 strict 10.5.1.1
 strict 10.5.1.2
path mu1_to_mu2_1
 strict 10.1.1.1
 strict 10.1.1.2
lsp frr1
 to 10.4.2.1
 cos 6
ipmtu 1028
traffic-eng max-rate 180 mean-rate 125
metric 5
enable

lsp lsp13d
 to 10.3.3.2
 primary mu1_to_mu3
 cos 7
 traffic-eng max-rate 250 mean-rate 120 no cspf
 enable

lsp lsp12d
 to 10.1.1.2
 cos 7
 traffic-eng max-rate 100 mean-rate 50
 enable
```

```
end of MPLS configuration
```

Displaying filtered MPLS configuration information

An individual MPLS interface, LSP, VLL, or VPLS can be specified in the **show run router mpls** command to display configuration of the specified object only. The following example displays the MPLS configuration information for the LSP named "frr1".

```
device# show run router mpls lsp frr1
 lsp frr1
  to 10.4.2.1
  cos 6
  ipmtu 1028
  traffic-eng max-rate 180 mean-rate 125
  metric 5
  frr
    bandwidth 80
    hop-limit 55
  enable
```

When an option is used without a variable specified, the configuration parameters for the option are shown for all elements that match the option are displayed. For instance, in the following example the **lsp name** option is used without a specified *lsp-name* variable. Consequently, the display contains the configuration information for all three LSPs configured on the router.

```
device# show run router mpls lsp
lsp frr1
  to 10.4.2.1
  cos 6
  ipmtu 1028
  traffic-eng max-rate 180 mean-rate 125
  metric 5
  enable

lsp lsp13d
 to 10.3.3.2
 primary mu1_to_mu3
 cos 7
 traffic-eng max-rate 250 mean-rate 120
 no cspf
 enable
```

Displaying RSVP information

The user can display RSVP version information, the status of RSVP interfaces, RSVP session information, and RSVP statistics.

Displaying the status of RSVP interfaces

Use the show mpls rsvp interface command to display the status of RSVP on devices where it is enabled.

Displaying RSVP protocol packet statistics

The device constantly gathers RSVP statistics. RSVP statistics are collected from the time RSVP is enabled, as well as from the last time the RSVP statistics counters were cleared.

To clear the RSVP statics counters, use the following command:

```
device# clear mpls rsvp statistics
```

This command resets the counters listed under "since last clear" for the **show mpls rsvp interface detail** and the **show mpls rsvp statistics** commands.

An addition command that displays the same information is the **show mpls rsvp statistics** command.

```
device# show mpls rsvp statistics
Total    Since    last clear
PacketType      Sent     Received      Sent      Received
Path            4        4             4         4
Resv            4        4             4         4
PathErr         0        0             0         0
ResvErr         0        0             0         0
PathTear        0        0             0         0
ResvTear        0        0             0         0
ResvConf        0        0             0         0

Errors                                Total       Since    last    clear
Rcv pkt    bad    length              0           0
Rcv    pkt    unknown type           0           0
Rcv    pkt    bad version            0           0
Rcv    pkt    bad cksum              0           0
Memory alloc fail              0           0
Rcv pkt processing error:
Path                           0           0
Resv                           0           0
PathErr                        0           0
ResvErr                        0           0
PathTear                       0           0
ResvTear                       0           0
ResvConf                       0           0
```

Displaying the RSVP version

To display the RSVP version number, as well as the refresh interval and refresh multiple, use the **show mpls rsvp** command.

MPLS traffic statistics

SLX-OS provides a mechanism to collect traffic statistics for MPLS entities. From a forwarding perspective, there are two kinds of entities, tunnel, and cross-connects.

For tunnels, the statistics refer to the number of packets or bytes pushed into the MPLS tunnel. It could be any traffic, such as IPoMPLS or VPLS.

Cross-connects are entities programmed on transit routers for forwarding labeled traffic. For a cross-connect, traffic is measured as the number of packets coming into the router with the particular label for which the cross-connect is programmed.

Tunnel statistics

The SLX-OS platform provides a mechanism to collect traffic statistics for MPLS entities. .

From a forwarding perspective, there are two kinds of entities, tunnel and cross-connects.

For tunnels, the statistics refer to the number of packets or bytes pushed into the MPLS tunnel. It could be any kind of traffic, such as IPoMPLS or VPLS.

Cross-connects are entities programmed on transit routers for forwarding labeled traffic. For a cross-connect, traffic is measured as the number of packets coming into the router with the particular label for which the cross-connect is programmed.

Ingress tunnel accounting

Ingress tunnel accounting provides the ability to count the number of traffic bytes and packets forwarded through a specified LSP. Ingress tunnel accounting is supports the following:

- RSVP-signaled LSPs
- LDP signaled LSPs

Configuring ingress tunnel accounting

This section explains how to configure ingress tunnel accounting at Link Switch Routers (LSR).

1. Enter **router mpls** command to configure MPLS in global configuration mode.
2. Enter **policy** command to set the MPLS policy.
3. Enter **ingress-tunnel-accounting** command to configure ingress tunnel accounting at Link Switch Routers (LSR).

The following example shows how to configure ingress tunnel accounting at Link Switch Routers (LSR).

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# ingress-tunnel-accounting
```

Displaying MPLS tunnel statistics

Displays the statistics of the MPLS tunnel.

The following example shows the output of the **show mpls statistics tunnel** command.

```
device# show mpls statistics tunnel
                              Total                 Since Last clear       Rate
Name      Intf       Prot    Packets    Bytes      Packets    Bytes       Packets/sec   Bytes/sec
t1        tnl0       R          2004    28175882   2004       28175882    6             93919
t2        tnl1       L          3101    40373763   3101       40373763    10            134579
```

Clearing MPLS tunnel statistics

Clears the statistics for the specified MPLS tunnel.

To clear the MPLS tunnel statistics for tunnel *1*, enter the following command.

```
device# clear mpls statistics tunnel 1
```

NOTE

Clearing the statistics does not clear the "Total" columns but does clear the "Since Last Clear" column.

Configuring transit session accounting

Configures the transit session accounting between labels, and well as LDP and RSVP.

To configure transit session accounting, complete the following steps.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enter the policy mode.

```
device(config-router-mpls)# policy
```

4. Configures the transit session accounting.

```
device(config-router-mpls-policy)# transit-session-accounting
```

The following example configures the transit session accounting.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# transit-session-accounting
```

Displaying MPLS transit statistics

Displays the statistics for the transit MPLS.

To display the transit MPLS statistics, enter the following command.

```
device# show mpls statistics transit
                    Total                Since Last clear     Rate
In Label    Prot    Packets  Bytes       Packets   Bytes      Packets/sec   Bytes/sec

2048        R       2004     28175882    2004      28175882   6             93919

2050        L       3101     40373763    3101      40373763   10            134579
```

Clearing MPLS transit statistics

Clears the transit statistics for a label.

In this example, to clear the transit statistics for label 2048, use the following command.

```
device# clear mpls statistics transit loabel 2048
```

NOTE
Clearing the statistics does not clear the "Total" columns but does clear the "Since Last Clear" column.

Adaptive Fast Reroute (FRR) and Global Revertiveness

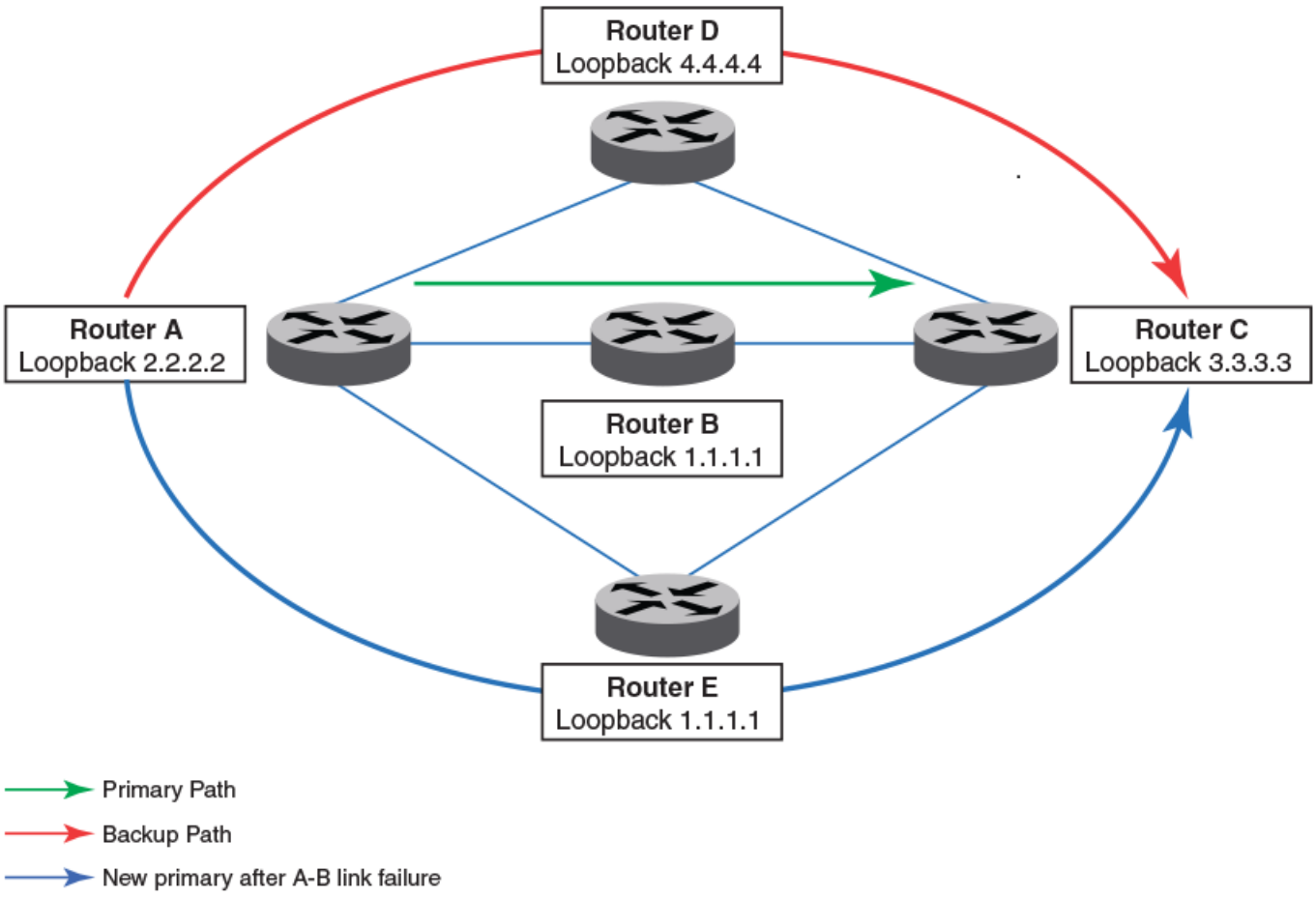
Adaptive capabilities support to Fast Reroute (FRR) and enabling global revertiveness enables the following capabilities:

- Once FRR is triggered, a make-before-break operation is performed to re-establish the primary path. When an established path attempts to reroute onto a new path, the ingress device continues to maintain existing paths and allocated bandwidths, ensuring that the existing path is not prematurely torn down and allowing the current traffic to continue flowing while the new path is set up.
- Configuration of the secondary path to have the LSP re-trigger the primary path is no longer required.
- The LSP waits for the configured revertive hold time after FRR is triggered before trying to re-optimize.

The following diagram shows an example of a primary LSP between A-B-C and backup over bypass tunnel on the path A-D-C. The primary LSP is configured without a strict path. When the interface between A-B goes down, the global revertiveness feature triggers a new LSP on the path A-E-C. The traffic is shifted to the new instance and old instance is torn down.

When the primary LSP is triggered with strict a path (A-B-C), after global revertiveness is triggered, a new instance tries the same path given in the strict path. In the diagram below, the new instance also tries to come up in the path A-B-C.

FIGURE 24 Sample topology for global revertiveness



Configuring FRR on an LSP to be adaptive

When an FRR is enabled, the user can change the following parameters without disabling the LSP:

- bandwidth
- exclude-any
- hop-limit
- include-all
- include-any
- priority

For instructions on how to configure an adaptive FRR LSP, refer to Configuring MPLS Fast Reroute using one-to-one backup on page 174.

Global Revertiveness

NOTE
Local revertiveness is not supported in this release.

When failover happens, traffic continues to flow in backup. When global revertiveness for FRR is configured, a new LSP is created from the ingress after the ingress learns about the failover. The new LSP is protected with a backup LSP, if possible. When the primary LSP fails for the second time, it may still be protected when there is a backup path available.

When secondary path is configured along with global revertive configuration, then when new instance of global revertive is triggered, the secondary path is also triggered. After "n" number of retries configured by user for establishing new instance for global revertiveness, traffic switches to the secondary path. The retry limit is configured in **mpls policy** mode. When the retry limit is not configured, then new instance establishment is tried infinite times.

Global revertiveness configuration

Global revertiveness is enabled by default in FRR mode for an adaptive LSP.

Complete the following steps to configure global revertiveness.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable to configure policy parameters.

```
device(config-router-mpls)# policy
```

4. Set the LSPs connect maximum number of retries.

```
device(config-router-mpls-policy)# retry-limit 20
```

In this example, the maximum number of retries is set to 20.

5. Exit the current level.

```
device(config-router-mpls-policy)# exit
```

6. Set the signaled label switched path (LSP).

```
device(config-router-mpls)# lsp t1
```

In this example, the LSPs name is "t1".

7. Enable the LSP to be modified without disabling the LSP.

```
device(config-router-mpls-lsp-t1)# adaptive
```

8. Enable for fast reroute options.

```
device(config-router-mpls-lsp-t1)# frr
```

9. Configure the revertive hold time for the LSP.

```
device(config-router-mpls-lsp-t1-frr)# revertive holdtime 20
```

   In this example, the revertive hold time is set to 20.

10. Exit from the current level.

```
device(config-routr-mpls-lsp-t1-frr)# exit
```

11. Apply the parameter modifications to the LSP.

```
device(config-router-mpls-lsp-t1)# commit
```

The following example show the how the global revertiveness is enabled by default in the fast reroute mode for an adaptive LSP.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# retry-limit 20
device(config-router-mpls-policy)# exit
device(config-router-mpls)# lsp t1
device(config-router-mpls-lsp-t1)# adaptive
device(config-router-mpls-lsp-t1)# frr
device(config-router-mpls-lsp-t1-frr)# revertive holdtime 20
device(config-router-mpls-lsp-t1-frr)# exit
device(config-router-mpls-lsp-t1)# commit
```

Changing FRR bandwidth for an adaptive LSP

Complete the following steps to change the fast reroute bandwidth for an adaptive LSP.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Set the signaled label switched path (LSP).

```
device(config-router-mpls)#lsp t1
```

   In this example, the LSPs name is "t1".

4. Enable for fast reroute options.

```
device(config-router-mpls-lsp-t1)#frr
```

5. Set the bandwidth for the detour or backup LSP.

```
device(config-router-mpls-lsp-t1-frr)# bandwidth 1000
```

   In this example, the maximum bandwidth for the detour or backup LSP is 1000 kbits. per second.

6. Exit from the current level.

```
device(config-router-mpls-lsp-t1-frr)# exit
```

7. Apply the parameter modifications to the LSP.

```
device(config-router-mpls-lsp-t1)# commit
```

The following example shows how to change the fast reroute bandwidth for an adaptive LSP.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp t1
device(config-router-mpls-lsp-t1)# frr
device(config-router-mpls-lsp-t1-frr)# bandwidth 1000
device(config-router-mpls-lsp-t1-frr)# exit
device(config-router-mpls-lsp-t1)# commit
```

Setting the revertive hold time

Use the revertive hold-time command to specify the time the LSP holds before attempting a new path on the FRR LSP.

1. Enable the device and configure the terminal to the global configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Set the signaled switched path (LSP). In this example, the LSPs name is *temp*.

   ```
   device(config-router-mpls)# lsp temp
   ```

4. Enable the LSP to be adaptive.

   ```
   device(config-router-mpls-lsp-temp)# adaptive
   ```

   This enables the LSP to be modified without restarting.

5. Enable the ability to set fast reroute options.

   ```
   device(config-router-mpls-lsp-temp)# frr
   ```

6. Enable global revertive mode.

   ```
   device(config-router-mpls-lsp-temp-frr)# revertive mode global enable
   ```

The following example shows how to enable the LSP holds before attempting a new path on the FRR LSP.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp temp
device(config-router-mpls-lsp-temp)# adaptive
device(config-router-mpls-lsp-temp)# frr
device(config-router-mpls-lsp-temp-frr)# revertive mode global enable
```

Global revertiveness configurations

Global revertiveness is enabled by default in FRR mode for an adaptive LSP.

Changing FRR bandwidth for an adaptive LSP

Complete the following steps to change the fast reroute bandwidth for an adaptive LSP.

1. Configure the device.

   ```
   device# configure
   ```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Set the signaled label switched path (LSP).

```
device(config-router-mpls)#lsp t1
```

In this example, the LSPs name is "t1".

4. Enable for fast reroute options.

```
device(config-router-mpls-lsp-t1)#frr
```

5. Set the bandwidth for the detour or backup LSP.

```
device(config-router-mpls-lsp-t1-frr)# bandwidth 1000
```

In this example, the maximum bandwidth for the detour or backup LSP is 1000 kbits. per second.

6. Exit from the current level.

```
device(config-router-mpls-lsp-t1-frr)# exit
```

7. Apply the parameter modifications to the LSP.

```
device(config-router-mpls-lsp-t1)# commit
```

The following example shows how to change the fast reroute bandwidth for an adaptive LSP.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp t1
device(config-router-mpls-lsp-t1)# frr
device(config-router-mpls-lsp-t1-frr)# bandwidth 1000
device(config-router-mpls-lsp-t1-frr)# exit
device(config-router-mpls-lsp-t1)# commit
```

Adaptive LSP configuration

Global revertiveness is enabled by default in FRR mode for an adaptive LSP.

Complete the following steps to configure an adaptive LSP.

1. Configure the device.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Set the signaled label switched path (LSP).

```
device(config-router-mpls)# lsp t1
```

4. Set the egress router of the LSP.

```
device(config-router-mpls-lsp-t1)# to 10.3.3.3
```

In this example, the IP address of the egress router is 10.3.3.3.

5. Set the ingress router of the LSP.

```
device(config-router-mpls-lsp-t1)# from 10.2.2.2
```

   In this example, the IP address of the ingress router is 10.2.2.2.

6. Set tie traffic engineering parameters mean rate in kbits. per second.

```
device(config-router-mpls-lsp-t1)# traffic-eng mean-rate 1000
```

   In this example, the mean rate is set to 1000 kbits. per second.

7. Enable the LSP to be modified without disabling the LSP.

```
device(config-router-mpls-lsp-t1)# adaptive
```

8. Set-up for fast reroute options.

```
device(config-router-mpls-lsp-t1)# frr
```

9. Exit the current mode.

```
device(config-router-mpls-lsp-t1-frr)# exit
```

The following example shows how to configure an LSP to be adaptive.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp t1
device(config-router-mpls-lsp-t1)# to 10.3.3.3
device(config-router-mpls-lsp-t1)# from 10.2.2.2
device(config-router-mpls-lsp-t1)# traffic-eng mean-rate 1000
device(config-router-mpls-lsp-t1)# adaptive
device(config-router-mpls-lsp-t1)# frr
device(config-router-mpls-lsp-t1-frr)# exit
device(config-router-mpls-lsp-t1)# enable
```

Displaying global revertiveness information

Use the **show mpls lsp name** *lsp_name* command to display revertive mode information. The **show mpls lsp name***lsp_name* command displays detailed information about a specific LSP name.

```
device# show mpls lsp name tunnel1
LSP tunnel1, to 10.3.3.3
  From: 10.2.2.2, admin: UP, status: UP, tunnel interface(primary path): tnl0
  Times primary LSP goes up since enabled: 1
  Metric: 0, number of installed aliases: 0  Adaptive
  Maximum retries: NONE, no. of retries: 0
  Pri. path: p1, up: yes (backup), active: yes
   Setup priority: 7, hold priority: 0
   Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
...
Active Path attributes:
   Tunnel interface: tnl0, outbound interface: e1/3
...
Backup LSP: UP, out-label: 3, outbound interface: e1/3
  Path cspf-group computation-mode: disabled
   Global revertiveness enabled with hold time 20 secs
   Revertive timer expires in 17 seconds
   FRR Forwarding State: Pri(down), Backup(active)
```

The output from the **show mpls lsp name** *lsp_name* command is enhanced to display the global revertiveness configuration. In this example, the global revertiveness is enabled with a hold time 20 seconds. The revertive timer is set to expire in 17 seconds. The secondary switchover timer is set to expire in 31 seconds which triggers the secondary path establishment.

MPLS over virtual Ethernet interfaces

Extreme devices support MPLS over *virtual ethernet (VE)* interfaces. MPLS over VE interfaces enables MPLS to be configured over tagged links. With this feature, MPLS can run over a single tag on the port. Other tags on the port can be used for other applications, such as Layer 2 VLANs, VPLS endpoints, and VLL endpoints.

An MPLS enabled VE interface supports the following services.

- IP over MPLS
- Transit LSR
- PBR over MPLS
- LSP Accounting
- MPLS VLL
- MPLS VPLS
- Multicast Snooping over VPLS
- 802.1ag
- MPLS OAM

NOTE
Multi-port static ARP configuration in not supported for MPLS uplinks.

Displaying MPLS configuration information for a VE interface

The **show running router mpls mpls-interface ve** *num* command to display specific MPLS interface configuration information. When MPLS is configured on a VE interface, the VE interface ID is displayed in the output.

The command allows the user to display configuration information for an MPLS-enabled interface. The user can specify a VE interface on the CLI. The following example displays CLI commands executed for the interface ve 20.

```
device# show running router mpls mpls-interface ve 20
  mpls-interface ve 20
  ldp-enable
```

MPLS enabled interface

When enabling MPLS on a VE interface, consider the following.

- The user cannot delete a VE interface while MPLS is enabled on it. The user must first remove MPLS from the interface configuration. The following error message is displayed.

```
device(config)# no interface ve 100
%%Error: MPLS is enabled on the interface. First disable MPLS on this interface.
```

- The user cannot delete a VLAN associated with a VE when MPLS is enabled on that VE. The user must first disable MPLS from the VE interface. The following error message is displayed.

```
device(config)# no vlan 20
Error - vlan can't be deleted as MPLS is enabled on associated VE interface
```

- When MPLS is enabled on an interface, the last IP address of a VE cannot be removed. The command is rejected. The following error message is displayed.

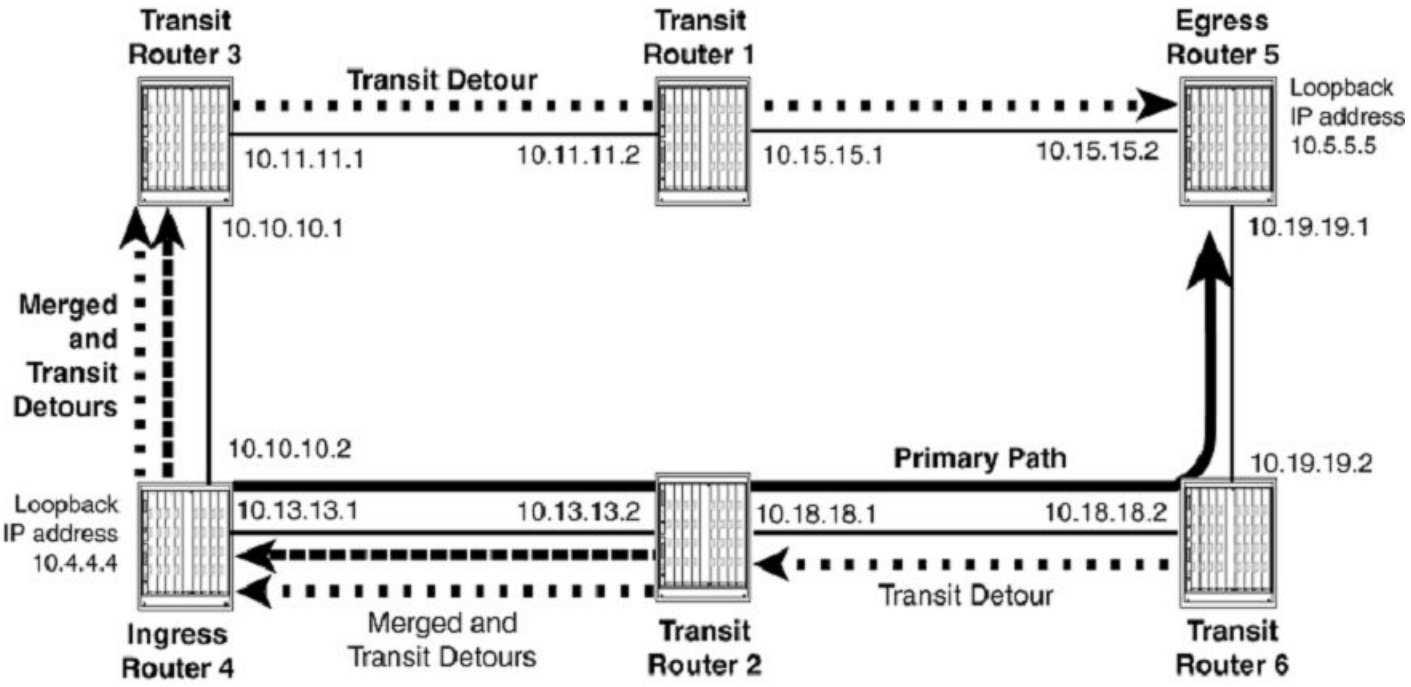
```
device(config-vif-54)# no ip address 10.40.40.5/24
IP/Port: Error(31) Can not remove IP address as MPLS is configured on the port
```

Example of MPLS Fast Reroute configuration

This example describes an MPLS Fast Reroute Loop Configuration. It provides the configuration required on Ingress Router 4 and examples of the **show mpls rsvp session** displays for all of the routers in the configuration. These examples show how the MPLS Fast Reroute configuration of an LSP affects the RSVP session on each of the routers in the configuration.

As illustrated in Figure 25 and described in the configuration example that follows, Ingress Router 4 is configured with a strict Label Switch Path A to Egress Router 5. In this configuration, when the path is broken between Ingress Router 4 and Egress Router 5, Transit Routers 2 or 6 take a detour path back through Ingress Router 4 and continue through Transit Routers 3 and 1 to reach Egress Router 5.

FIGURE 25 MPLS Fast Reroute Loop configuration



The following is the MPLS Fast Reroute configuration for Ingress Router 4.

```
device(config)# interface loopback 1
device4(config-Loopback-1)# ip address 10.4.4.4/24
device4(config)# interface ethernet 2/1
device4(config-if-eth-2/1)# ip address 10.10.10.2/24
device4(config)# interface ethernet 2/9
device4(config-if-eth-2/9)# ip address 10.13.13.1/24
device4(config)# router mpls
device4(config-router-mpls)# mpls-interface ethernet 2/1 ethernet 2/9
device4(config-router-mpls-if-eth-2/1-eth-2/9)# path a
device4(config-router-mpls-path-a)#hop 10.2.2.2 strict
device4(config-router-mpls-path-a)# hop 10.6.6.6 strict
device4(config-router-mpls)# lsp 1
device4(config-router-mpls-lsp-1)# to 10.5.5.5
device4(config-router-mpls-lsp-1)# primary-path a
device4(config-router-mpls-lsp-1)# frr
```

Displaying RSVP session information for example network

The **show mpls rsvp session** command, provides information regarding the primary and detour routes in an MPLS RSVP fast reroute enabled network.

Display examples are provided for the following routers in the configuration shown in the preceding MPLS fast reroute configuration diagram.

- Transit router 6
- Transit router 2
- Ingress router 4
- Transit router 3
- Transit router 1

- Egress router 5

The following examples include displays for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

The transit router1 display

The following display examples are from transit router1. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

The display for transit router1 shows a transit detour (DT) path. The DT path is the detour path that is an alternative to the primary path configured on ingress router4 from ingress router4 to egress router5. This detour path shown from ingress router4 at loopback IP address *10.4.4.4* to egress router5 at loopback IP address *10.5.5.5* is only used when there is a failed link or router between the source and destination of the primary path.

```
device1# show mpls rsvp session
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP:   0 session(s)
Transit RSVP:   1 session(s)
To        From         St Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DT) Up SE     1028    3        1
Egress RSVP: 0 session(s)
```

The following example displays the output from transit router1 using the **show mpls rsvp session detail** command. This option provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DT path, the "Explicit path hop count" field indicates that there is one hop from this router to the egress of the path at the router at IP address *10.15.15.2* (egress router5).

```
device3# show mpls rsvp session detail
Ingress RSVP: 0 session(s)
Transit RSVP: 1 session(s)
To        From         St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DT) Up  SE     1028    3        1
 Time left in seconds (PATH refresh: 18, ttd: 146
 RESV refresh: 15, ttd: 154)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Explicit path hop count: 1
  10.15.15.2 (S)
 Received RRO count: 1
 Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.15.15.2
 PATH rcvfrom: 10.11.11.1 (e7/16) (MD5 OFF)
 PATH sentto:  10.15.15.2 (e6/2 ) (MD5 OFF)
 RESV rcvfrom: 10.15.15.2 (e6/2 ) (MD5 OFF)
Egress RSVP: 0 session(s)
```

The transit router2 display

The following display examples are from transit router2. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

As for transit router2, the displays show an ingress detour (DI) path, and a path without a code which identifies a protected path. In addition, a merged detour (DM) path is shown. The DM path is the detour path merged from transit router6. All three paths are shown from ingress router4 at loopback IP address *10.4.4.4* to egress router 5at loopback IP address *10.5.5.5* .

```
device1# show mpls rsvp session
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP:   0 session(s)
Transit RSVP:   1 session(s)
To        From         St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DI) Up  SE     1024    1024     1
```

```
10.5.5.5  10.4.4.4     Up  SE     1024    1024     1
10.5.5.5  10.4.4.4(DM) Up  SE     1025    1024     1
Egress RSVP: 0 session(s)
```

The following example displays the output from transit router2 using the **show mpls rsvp session detail** command. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DI path, two PLR and avoid node ID pairs are shown labeled [1] and [2]. In [1] the point of local repair (PLR) is at IP address *10.18.18.1* which is the interface to the primary path on transit router2 and the avoid node is IP address *10.18.18.2* on transit router6. In [2] the PLR is at IP address *10.19.19.2* on transit router6 and the avoid node is IP address *0.0.0.0* . The "Explicit path hop count" field indicates that there are four hops on this path from this router to the egress of the path at routers with the following IP addresses *10.13.13.1* (ingress router4), *10.10.10.1* (transit router3), *10.11.11.2* (transit router1), and *10.15.15.2* (egress router5).

For the DM path, one PLR and avoid node ID pair is shown labeled [1]. In [1], the PLR is at IP address *10.19.19.2* which is an interface on transit router6 and the avoid node is IP address *0.0.0.0* .

For the primary path, the "Explicit path hop count" field indicates that the path has two hops from this router to the egress to the path at routers with the following IP addresses *10.18.18.2* (transit router6) and *10.19.19.1* (egress router5). The 'Fast Reroute' field indicates that the primary path has been configured for one-to-one backup.

```
device2# show mpls rsvp session detail
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP: 0 session(s)
Transit RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DI)  Up  SE     1024    1024     1
 Time left in seconds (PATH refresh: 1, ttd: 4293570
                       RESV refresh: 13, ttd: 141)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Explicit path hop count: 4
  10.13.13.1 (S) -> 10.10.10.1 (S) -> 10.11.11.2 (S) -> 10.15.15.2 (S)
 Received RRO count: 4
  Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.13.13.1 -> 10.10.10.1 -> 10.11.11.2 ->
   10.15.15.2
 Detour Sent: Number of PLR and Avoid Node ID pair(s): 2
  [1]: PLR: 10.18.18.1 Avoid Node: 10.18.18.2
  [2]: PLR: 10.19.19.2 Avoid Node: 0.0.0.0
 PATH sentto: 10.13.13.1  (e5/10 ) (MD5 OFF)
 RESV rcvfrom: 10.13.13.1 (e5/10 ) (MD5 OFF)
To        From      St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4  Up  SE     1024    1024     1
 Time left in seconds (PATH refresh: 26, ttd: 151
                       RESV refresh: 13, ttd: 151)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Fast Reroute: one-to-one backup desired
  Setup priority: 7, hold priority: 0
  Bandwidth: 0 kbps, hop limit: 255
  Detour LSP: UP. Nexthop (node) protection available.
   Up/Down times: 1, num retries: 0
 Explicit path hop count: 2
  10.18.18.2 (S) -> 10.19.19.1 (S)
 Received RRO count: 2
  Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.18.18.2 (PN) -> 10.19.19.1
 PATH rcvfrom: 10.13.13.1 (e5/10 ) (MD5 OFF)
 PATH sentto: 10.18.18.2  (e2/1 )  (MD5 OFF)
 RESV rcvfrom: 10.18.18.2 (e2/1 )  (MD5 OFF)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DM)  Up  SE     1025    1024     1
 Time left in seconds (PATH refresh: 31, ttd: 133
                       RESV refresh: 13, ttd: 141)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Received RRO count: 4
  Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.13.13.1 -> 10.10.10.1 -> 10.11.11.2 ->
```

```
  10.15.15.2
Detour Rcvd: Number of PLR and Avoid Node ID pair(s): 1
 [1]: PLR: 10.19.19.2 Avoid Node: 0.0.0.0
PATH rcvfrom: 10.18.18.2 (e2/1 ) (MD5 OFF)
RESV rcvfrom: 10.13.13.1 (e5/10 ) (MD5 OFF)
Egress RSVP: 0 session(s)
```

The transit router3 display

The following display examples come from transit router3. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

The display for transit router3 shows a transit detour (DT) path. The DT path is the detour path that is an alternative to the primary path configured on ingress router4 from itself to egress router5. This detour path, shown from ingress router4 at loopback IP address *10.4.4.4* to egress router5 at loopback IP address *10.5.5.5* , is only used when a link or router fails between the source and destination of the primary path.

```
device3# show mpls rsvp session
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP: 0 session(s)
Transit RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DT)  Up  SE     1028    1028     1
Egress RSVP: 0 session(s)
```

The following example displays the output from transit router3 using the **show mpls rsvp session detail** command. This command provides additional details about the paths described in the output from the **show mpls rsvp** *session* command.

For the DT path, the "Explicit path hop count:" field shows that two hops exist from this router to the egress of the path at routers with the following at IP addresses: *10.11.11.2* (transit router1) and *10.15.15.2* (egress router5).

```
device3# show mpls rsvp session detail
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP: 0 session(s)
Transit RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DT)  Up  SE     1028    1028     1
 Time left in seconds (PATH refresh: 2, ttd: 141
 RESV refresh: 25, ttd: 154)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Explicit path hop count: 2
  10.11.11.2 (S) -> 10.15.15.2 (S)
 Received RRO count: 2
 Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.11.11.2 -> 10.15.15.2
 PATH rcvfrom: 10.10.10.2 (e12/1 )  (MD5 OFF)
 PATH sentto:  10.11.11.2 (e11/18 ) (MD5 OFF)
 RESV rcvfrom: 10.11.11.2 (e11/18 ) (MD5 OFF)
Egress RSVP: 0 session(s)
```

The ingress router4 display

The following display examples are from ingress router4. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

Similar to the display for transit router2, the displays show an ingress detour (DI) path, a path without a code which identifies a protected path and merged detour (DM) path. The DM path is the detour path merged from transit routers 2 and 6. All three paths are shown from ingress router4 at IP address loopback *10.4.4.4* to egress router5 at IP address loopback *10.5.5.5* . In the case of the DM path, a

reroute at either transit router 2 or 6 sends traffic that had begun at ingress router4 back through it, and forward through transit routers 3 and 1 to the ultimate destination at egress router5.

```
device4# show mpls rsvp session
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DI)  Up  SE     -       1028     1
10.5.5.5  10.4.4.4(DM)  Up  SE     1024    1028     1
10.5.5.5  10.4.4.4      Up  SE     -       1024     1
Transit RSVP: 0 session(s)
Egress RSVP: 0 session(s)
```

The following example displays the **show mpls rsvp session detail** output for ingress router4. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DI path, three PLR and avoid node ID pairs are shown labeled [1], [2] and [3]. In [1] and [2] the PLR is at IP Address *10.13.13.1* which is the interface to the primary path on ingress router4. The avoid node for pair [1] is IP address *10.13.13.2* on transit router2 and the avoid node for pair [2] is IP address *10.18.18.2* on transit router6. In [3] the PLR is at IP address *10.18.18.1* which is the interface to the primary path on ingress router2 and the avoid node for pair [3] is IP address *10.18.18.2* on transit router6. The "Explicit path hop count" field indicates that there are three hops on the path from this router to the egress of the path at routers with the following IP addresses *10.10.10.1* (transit router3), *10.11.11.2* (transit router1) and *10.15.15.2* (egress router5).

For the DM path, one PLR and avoid node ID pair is shown labeled [1]. In [1] the PLR is at IP address *10.18.18.1* which is the interface to the primary path on ingress router2 and the avoid node is IP address *10.18.18.2* on transit router6.

For the primary path, the "Explicit path hop count" field indicates that there are three hops on this path from ingress router4 to the egress to the path at routers with the following IP addresses *10.13.13.2* (transit router2), *10.18.18.2* (transit router6) and *10.19.19.1* (egress router5). The "Fast Reroute" field indicates that the primary path has been configured for one-to-one backup.

```
device4# show mpls rsvp session detail
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DI)  Up  SE     -       1028n    1
 Time left in seconds (PATH refresh: 16, ttd: 4293608
                       RESV refresh: 27, ttd: 133)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 65535
 Explicit path hop count: 3
  10.10.10.1 (S) -> 10.11.11.2 (S) -> 10.15.15.2 (S)
 Received RRO count: 3
  Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.10.10.1 -> 10.11.11.2 -> 10.15.15.2
 Detour Sent: Number of PLR and Avoid Node ID pair(s): 3
  [1]: PLR: 10.13.13.1 Avoid Node: 10.13.13.2
  [2]: PLR: 10.13.13.1 Avoid Node: 10.18.18.2
  [3]: PLR: 10.18.18.1 Avoid Node: 10.18.18.2
 PATH sentto: 10.10.10.1 (e2/1 ) (MD5 OFF)
 RESV rcvfrom: 10.10.10.1 (e2/1 ) (MD5 OFF)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DM)  Up  SE     1024    1028     1
 Time left in seconds (PATH refresh: 6, ttd: 134
                       RESV refresh: 27, ttd: 133)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Received RRO count: 3
  Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.10.10.1 -> 10.11.11.2 -> 10.15.15.2
 Detour Rcvd: Number of PLR and Avoid Node ID pair(s): 1
  [1]: PLR: 10.18.18.1 Avoid Node: 10.18.18.2
 PATH rcvfrom: 10.13.13.2 (e2/20 ) (MD5 OFF)
 RESV rcvfrom: 10.10.10.1 (e2/1 ) (MD5 OFF)
To        From      St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4  Up  SE     -       1024     1
 Time left in seconds (PATH refresh: 37, ttd: 148
```

```
                      RESV refresh: 27, ttd: 152)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 65535
 Fast Reroute: one-to-one backup desired
  Setup priority: 7, hold priority: 0
  Bandwidth: 0 kbps, hop limit: 255
  Detour LSP: UP. Nexthop (node) protection available.
   Up/Down times: 1, num retries: 0
 Explicit path hop count: 3
  10.13.13.2 (S) -> 10.18.18.2 (S) -> 10.19.19.1 (S)
 Received RRO count: 3
  Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.13.13.2 (PN) -> 10.18.18.2 (PN) -> 10.19.19.1
 PATH sentto: 10.13.13.2 (e2/20 ) (MD5 OFF)
 RESV rcvfrom: 10.13.13.2 (e2/20 ) (MD5 OFF)
Transit RSVP: 0 session(s)
Egress RSVP: 0 session(s)
```

The egress router5 display

The following display examples are from egress router5. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

The display for egress router5 shows an egress detour (DE) path and a path without a code that identifies a protected path. Both paths are shown from ingress router4 at loopback IP address *10.4.4.4* to egress router5 at loopback IP address *10.5.5.5* . The primary path traverses transit routers 2 and 6. In the case of the DE path, a reroute sends traffic through transit routers 3 and 1.

```
device5# show mpls rsvp session
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP: 0 session(s)
Transit RSVP: 0 session(s)
Egress RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DE)  Up  SE     3       0        1
10.5.5.5  10.4.4.4      Up  SE     3       0        1
```

The following example displays the output from egress router5 using the **show mpls rsvp session detail** command. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DE path, two PLR and avoid node ID pairs are shown labeled [1] and [2]. In both the PLR is at IP address *10.13.13.1* which is the interface to the primary path on ingress router4. The avoid node for pair [1] is IP address *10.13.13.2* on transit router2 and the avoid node for pair [2] is IP address *10.18.18.2* on transit router6.

The "Fast Reroute" field indicates that the primary path has been configured for one-to-one backup.

There is no "Explicit path hop count" field for either route because egress router 5 is the destination of the path.

```
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
               DE:Egress Detour RP:Repaired Session
Ingress RSVP: 0 session(s)
Transit RSVP: 0 session(s)
Egress RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DE)  Up  SE     3       0        1
 Time left in seconds (PATH refresh: 18, ttd: 149
                      RESV refresh: 7, ttd: 152)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Detour Rcvd: Number of PLR and Avoid Node ID pair(s): 2
  [1]: PLR: 10.13.13.1 Avoid Node: 10.13.13.2
  [2]: PLR: 10.13.13.1 Avoid Node: 10.18.18.2
 PATH rcvfrom: 10.15.15.1 (e8/2 ) (MD5 OFF)
To        From      St  Styl  Lbl_in  Lbl_outm  LSPname
10.5.5.5  10.4.4.4  Up  SE    3       0         1
 Time left in seconds (PATH refresh: 30, ttd: 152
                      RESV refresh: 7, ttd: 152)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
```

```
 Fast Reroute: one-to-one backup desired
  Setup priority: 7, hold priority: 0
  Bandwidth: 0 kbps, hop limit: 255
 PATH rcvfrom: 10.19.19.2 (e8/1) (MD5 OFF)
```

The transit router6 display

The following display examples are from transit router6. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

Both displays show two paths from ingress router4 at loopback IP address *10.4.4.4* to egress router5 at loopback IP address *10.5.5.5* . The (DI) path is an ingress detour path, and the path without a code is a protected path. The DI path is the detour path that is taken when transit router6 is unable to use the primary path to egress router5.

```
device6# show mpls rsvp session
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP: 0 session(s)
Transit RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DI)  Up  SE     1024    1025     1
10.5.5.5  10.4.4.4      Up  SE     1024    3        1
Egress RSVP: 0 session(s)
```

The following example displays the output from transit router6 using the **show mpls rsvp session detail** command. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DI path, one PLR and avoid node ID pair is shown labeled [1]. In [1], the PLR is at IP address *10.19.19.2* which is an interface on transit router6 and the avoid node is IP address *0.0.0.0* . The "Explicit path hop count" field indicates that there are five hops on this path from this router to the egress to the path at routers with the following IP addresses *10.18.18.1* (transit router2), *10.13.13.1* (ingress router4), *10.10.10.1* (transit router3), *10.11.11.2* (transit router1) and *10.15.15.2* (egress router5).

For the primary path, the "Explicit path hop count" field indicates that there is one hop on this path from this router to the egress to the path to the router at IP address *10.19.19.1* (egress router5). The "Fast Reroute" field indicates that the primary path has been configured for one-to-one backup.

```
device6# show mpls rsvp session detail
Codes: DI:Ingress Detour DT:Transit Detour DM:Merged Detour
       DE:Egress Detour RP:Repaired Session
Ingress RSVP: 0 session(s)
Transit RSVP: 1 session(s)
To        From          St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4(DI)  Up  SE     1024    1025     1
 Time left in seconds (PATH refresh: 6, ttd: 4293497
 RESV refresh: 24, ttd: 131)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Explicit path hop count: 5
  10.18.18.1 (S) -> 10.13.13.1 (S) -> 10.10.10.1 (S) -> 10.11.11.2 (S) ->
  10.15.15.2 (S)
 Received RRO count: 5
  Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.18.18.1 -> 10.13.13.1 -> 10.10.10.1 ->
  10.11.11.2 -> 10.15.15.2
 Detour Sent: Number of PLR and Avoid Node ID pair(s): 1
  [1]: PLR: 10.19.19.2 Avoid Node: 0.0.0.0
 PATH sentto: 10.18.18.1  (e5/1 ) (MD5 OFF)
 RESV rcvfrom: 10.18.18.1 (e5/1 ) (MD5 OFF)
To        From      St  Style  Lbl_in  Lbl_out  LSPname
10.5.5.5  10.4.4.4  Up  SE     1024    3        1
 Time left in seconds (PATH refresh: 28, ttd: 150
                       RESV refresh: 24, ttd: 128)
 Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
 Fast Reroute: one-to-one backup desired
  Setup priority: 7, hold priority: 0
  Bandwidth: 0 kbps, hop limit: 255
```

```
 Detour LSP: UP. Nexthop (node) protection available.
  Up/Down times: 1, num retries: 0
Explicit path hop count: 1
 10.19.19.1 (S)
Received RRO count: 1
 Protection codes: P: Local N: Node B: Bandwidth I: InUse
  10.19.19.1
PATH rcvfrom: 10.18.18.1 (e5/1 ) (MD5 OFF)
PATH sentto: 10.19.19.1  (e5/2 ) (MD5 OFF)
RESV rcvfrom: 10.19.19.1 (e5/2 ) (MD5 OFF)
Egress RSVP: 0 session(s)
```

The Transit Router 1 display

The following display examples are from Transit Router 1. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

The display for Transit Router 1 shows a *Transit Detour (DT)* path. The DT path is the detour path that is an alternative to the primary path configured on Ingress Router 4 from Ingress Router 4 to Egress Router 5. This detour path shown from Ingress Router 4 at Loopback IP address 10.4.4.4 to Egress Router 5 at Loopback IP address 10.5.5.5 is only used when there is a failed link or router between the source and destination of the primary path.

```
device1# show mpls rsvp session
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
Transit RSVP:     1 session(s)
To                From             St    Style Lbl_in Lbl_out LSPname
10.5.5.5          10.4.4.4(DT)     Up    SE    1028   3       1
Egress RSVP:      0 session(s)
```

The following example displays the output from Transit Router 1 using the **show mpls rsvp session detail** command. This option provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DT path, the "Explicit path hop count" field indicates that there is one hop from this router to the egress of the path at the router at IP address 10.15.15.2 (Egress Router 5).

```
device3# show mpls rsvp session detail
Ingress RSVP:     0 session(s)
Transit RSVP:     1 session(s)
To                From             St    Style Lbl_in Lbl_out LSPname
10.5.5.5          10.4.4.4(DT)     Up    SE    1028   3       1
  Time left in seconds (PATH refresh: 18, ttd: 146
                        RESV refresh: 15, ttd: 154)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Explicit path hop count: 1
   10.15.15.2 (S)
  Received RRO count: 1
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.15.15.2
  PATH rcvfrom: 10.11.11.1      (e7/16         ) (MD5 OFF)
  PATH sentto:  10.15.15.2      (e6/2          ) (MD5 OFF)
  RESV rcvfrom: 10.15.15.2      (e6/2          ) (MD5 OFF)
Egress RSVP:      0 session(s)
```

The Ingress Router 4 display

The following display examples are from Ingress Router 4. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

Like the display for Transit Router 2, the displays show an *Ingress Detour (DI)* path, a path without a code which identifies a protected path and *Merged Detour (DM)* path. The DM path is the detour path merged from Transit Routers 2 and 6. All three paths are shown

from Ingress Router 4 at IP address Loopback 10.4.4.4 to Egress Router 5 at IP address Loopback 10.5.5.5. In the case of the DM path, a reroute at either Transit Router 2 or 6 sends traffic that had begun at Ingress Router 4 back through it, and forward through Transit Routers 3 and 1 to the ultimate destination at Egress Router 5.

```
device4# show mpls rsvp session
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     1 session(s)
To                From              St   Style Lbl_in Lbl_out LSPname
10.5.5.5          10.4.4.4(DI)      Up   SE    -      1028    1
10.5.5.5          10.4.4.4(DM)      Up   SE    1024   1028    1
10.5.5.5          10.4.4.4          Up   SE    -      1024    1
Transit RSVP:     0 session(s)
Egress RSVP:      0 session(s)
```

The following example displays the **show mpls rsvp session detail** output for Ingress Router 4. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DI path, three PLR and Avoid Node ID pairs are shown labeled [1], [2] and [3]. In [1] and [2] the *Point of Local Repair (PLR)* is at IP Address 10.13.13.1 which is the interface to the primary path on Ingress Router 4. The Avoid Node for pair [1] is IP address 10.13.13.2 on Transit Router 2 and the Avoid Node for pair [2] is IP address 10.18.18.2 on Transit Router 6. In [3] the *Point of Local Repair (PLR)* is at IP Address 10.18.18.1 which is the interface to the primary path on Ingress Router 2 and the Avoid Node for pair [3] is IP address 10.18.18.2 on Transit Router 6. The "Explicit path hop count" field indicates that there are three hops on the path from this router to the egress of the path at routers with the following IP addresses 10.10.10.1 (Transit Router 3), 10.11.11.2 (Transit Router 1) and 10.15.15.2 (Egress Router 5).

For the DM path, one PLR and Avoid Node ID pair is shown labeled [1]. In [1] the *Point of Local Repair (PLR)* is at IP Address 10.18.18.1 which is the interface to the primary path on Ingress Router 2 and the Avoid Node is IP address 10.18.18.2 on Transit Router 6.

For the primary path, the "Explicit path hop count" field indicates that there are three hops on this path from Ingress Router 4 to the egress to the path at routers with the following IP addresses 10.13.13.2 (Transit Router 2), 10.18.18.2 (Transit Router 6) and 10.19.19.1 (Egress Router 5). The "Fast Reroute" field indicates that the primary path has been configured for one-to-one backup.

```
device4# show mpls rsvp session detail
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     1 session(s)
To                From              St    Style Lbl_in Lbl_out LSPname
10.5.5.5          10.4.4.4(DI)      Up    SE    -      1028n   1
  Time left in seconds (PATH refresh: 16, ttd: 4293608
                        RESV refresh: 27, ttd: 133)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 65535
  Explicit path hop count: 3
   10.10.10.1 (S) -> 10.11.11.2 (S) -> 10.15.15.2 (S)
  Received RRO count: 3
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.10.10.1 -> 10.11.11.2 -> 10.15.15.2
  Detour Sent: Number of PLR and Avoid Node ID pair(s): 3
   [1]: PLR: 10.13.13.1  Avoid Node: 10.13.13.2
   [2]: PLR: 10.13.13.1  Avoid Node: 10.18.18.2
   [3]: PLR: 10.18.18.1  Avoid Node: 10.18.18.2
  PATH sentto:  10.10.10.1      (e2/1          ) (MD5 OFF)
  RESV rcvfrom: 10.10.10.1      (e2/1          ) (MD5 OFF)
To                From                 St    Style Lbl_in Lbl_out LSPname
10.5.5.5          10.4.4.4(DM)         Up    SE    1024   1028    1
  Time left in seconds (PATH refresh: 6, ttd: 134
                        RESV refresh: 27, ttd: 133)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Received RRO count: 3
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.10.10.1 -> 10.11.11.2 -> 10.15.15.2
  Detour Rcvd: Number of PLR and Avoid Node ID pair(s): 1
   [1]: PLR: 10.18.18.1  Avoid Node: 10.18.18.2
```

```
  PATH rcvfrom: 10.13.13.2      (e2/20         ) (MD5 OFF)
  RESV rcvfrom: 10.10.10.1      (e2/1          ) (MD5 OFF)
To               From            St    Style Lbl_in Lbl_out LSPname
10.5.5.5         10.4.4.4        Up    SE    -      1024    1
  Time left in seconds (PATH refresh: 37, ttd: 148
                        RESV refresh: 27, ttd: 152)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 65535
  Fast Reroute: one-to-one backup desired
   Setup priority: 7, hold priority: 0
   Bandwidth: 0 kbps, hop limit: 255
   Detour LSP: UP.  Nexthop (node) protection available.
    Up/Down times: 1, num retries: 0
  Explicit path hop count: 3
   10.13.13.2 (S) -> 10.18.18.2 (S) -> 10.19.19.1 (S)
  Received RRO count: 3
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.13.13.2 (PN) -> 10.18.18.2 (PN) -> 10.19.19.1
  PATH sentto:  10.13.13.2      (e2/20         ) (MD5 OFF)
  RESV rcvfrom: 10.13.13.2      (e2/20         ) (MD5 OFF)
Transit RSVP:     0 session(s)
Egress RSVP:      0 session(s)
```

The Transit Router 2 display

The following display examples are from Transit Router 2. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

As for Transit Router 2, the displays show an *Ingress Detour (DI)* path, and a path without a code which identifies a protected path. In addition, a *Merged Detour (DM)* path is shown. The DM path is the detour path merged from Transit Router 6. All three paths are shown from Ingress Router 4 at Loopback IP address 10.4.4.4 to Egress Router 5 at Loopback IP address 10.5.5.5.

```
device1# show mpls rsvp session
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
Transit RSVP:     1 session(s)
To               From              St  Style Lbl_in Lbl_out LSPname
10.5.5.5         10.4.4.4(DI)      Up  SE    1024   1024    1
10.5.5.5         10.4.4.4          Up  SE    1024   1024    1
10.5.5.5         10.4.4.4(DM)      Up  SE    1025   1024    1
Egress RSVP:      0 session(s)
```

The following example displays the output from Transit Router 2 using the **show mpls rsvp session detail** command. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DI path, two PLR and Avoid Node ID pairs are shown labeled [1] and [2]. In [1] the *Point of Local Repair (PLR)* is at IP Address 10.18.18.1 which is the interface to the primary path on Transit Router 2 and the Avoid Node is IP address 10.18.18.2 on Transit Router 6. In [2] the *Point of Local Repair (PLR)* is at IP Address 10.19.19.2 on Transit Router 6 and the Avoid Node is IP address 0.0.0.0. The "Explicit path hop count" field indicates that there are four hops on this path from this router to the egress of the path at routers with the following IP addresses 10.13.13.1 (Ingress Router 4), 10.10.10.1 (Transit Router 3), 10.11.11.2 (Transit Router 1) and 10.15.15.2 (Egress Router 5).

For the DM path, one PLR and Avoid Node ID pair is shown labeled [1]. In [1] the *Point of Local Repair (PLR)* is at IP Address 10.19.19.2 which is an interface on Transit Router 6 and the Avoid Node is IP address 0.0.0.0.

For the primary path, the "Explicit path hop count" field indicates that the path has two hops from this router to the egress to the path at routers with the following IP addresses 10.18.18.2 (Transit Router 6) and 10.19.19.1 (Egress Router 5). The 'Fast Reroute' field indicates that the primary path has been configured for one-to-one backup.

```
device2# show mpls rsvp session detail
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
```

```
Transit RSVP:     1 session(s)
To              From            St   Style Lbl_in Lbl_out LSPname
10.5.5.5        10.4.4.4(DI)    Up   SE    1024   1024    1
  Time left in seconds (PATH refresh: 1, ttd: 4293570
                        RESV refresh: 13, ttd: 141)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Explicit path hop count: 4
   10.13.13.1 (S) -> 10.10.10.1 (S) -> 10.11.11.2 (S) -> 10.15.15.2 (S)
  Received RRO count: 4
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.13.13.1 -> 10.10.10.1 -> 10.11.11.2 ->
    10.15.15.2
  Detour Sent: Number of PLR and Avoid Node ID pair(s): 2
   [1]: PLR: 10.18.18.1  Avoid Node: 10.18.18.2
   [2]: PLR: 10.19.19.2  Avoid Node: 0.0.0.0
  PATH sentto:  10.13.13.1      (e5/10         ) (MD5 OFF)
  RESV rcvfrom: 10.13.13.1      (e5/10         ) (MD5 OFF)
To              From             St   Style Lbl_in Lbl_out LSPname
10.5.5.5        10.4.4.4         Up   SE    1024   1024    1
  Time left in seconds (PATH refresh: 26, ttd: 151
                        RESV refresh: 13, ttd: 151)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Fast Reroute: one-to-one backup desired
   Setup priority: 7, hold priority: 0
   Bandwidth: 0 kbps, hop limit: 255
   Detour LSP: UP.  Nexthop (node) protection available.
    Up/Down times: 1, num retries: 0
  Explicit path hop count: 2
   10.18.18.2 (S) -> 10.19.19.1 (S)
  Received RRO count: 2
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.18.18.2 (PN) -> 10.19.19.1
  PATH rcvfrom: 10.13.13.1      (e5/10         ) (MD5 OFF)
  PATH sentto:  10.18.18.2      (e2/1          ) (MD5 OFF)
  RESV rcvfrom: 10.18.18.2      (e2/1          ) (MD5 OFF)
To                  From                    St    Style Lbl_in  Lbl_out  LSPname
10.5.5.5          10.4.4.4(DM)          Up    SE    1025    1024     1
  Time left in seconds (PATH refresh: 31, ttd: 133
                        RESV refresh: 13, ttd: 141)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Received RRO count: 4
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.13.13.1 -> 10.10.10.1 -> 10.11.11.2 ->
    10.15.15.2
  Detour Rcvd: Number of PLR and Avoid Node ID pair(s): 1
   [1]: PLR: 10.19.19.2  Avoid Node: 0.0.0.0
  PATH rcvfrom: 10.18.18.2      (e2/1          ) (MD5 OFF)
  RESV rcvfrom: 10.13.13.1      (e5/10         ) (MD5 OFF)
Egress RSVP:      0 session(s)
```

The Transit Router 3 display

The following display examples come from Transit Router 3. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

The display for Transit Router 3 shows a *Transit Detour (DT)* path. The DT path is the detour path that is an alternative to the primary path configured on Ingress Router 4 from itself to Egress Router 5. This detour path, shown from Ingress Router 4 at Loopback IP address 10.4.4.4 to Egress Router 5 at Loopback IP address 10.5.5.5, is only used when a link or router fails between the source and destination of the primary path.

```
device3# show mpls rsvp session
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
Transit RSVP:     1 session(s)
To              From             St    Style Lbl_in Lbl_out LSPname
```

```
10.5.5.5        10.4.4.4(DT)      Up    SE    1028   1028    1
Egress RSVP:      0 session(s)
```

The following example displays the output from Transit Router 3 using the **show mpls rsvp session detail** command. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DT path, the "Explicit path hop count:" field shows that two hops exist from this router to the egress of the path at routers with the following at IP addresses: 10.11.11.2 (Transit Router 1) and 10.15.15.2 (Egress Router 5).

```
device3# show mpls rsvp session detail
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
Transit RSVP:     1 session(s)
To              From             St    Style Lbl_in Lbl_out LSPname
10.5.5.5        10.4.4.4(DT)     Up    SE    1028   1028    1
  Time left in seconds (PATH refresh: 2, ttd: 141
                        RESV refresh: 25, ttd: 154)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Explicit path hop count: 2
   10.11.11.2 (S) -> 10.15.15.2 (S)
  Received RRO count: 2
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.11.11.2 -> 10.15.15.2
  PATH rcvfrom: 10.10.10.2      (e12/1         ) (MD5 OFF)
  PATH sentto:  10.11.11.2      (e11/18        ) (MD5 OFF)
  RESV rcvfrom: 10.11.11.2      (e11/18        ) (MD5 OFF)
Egress RSVP:      0 session(s)
```

The Egress Router 5 display

The following display examples are from Egress Router 5. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

The display for Egress Router 5, shows an *Egress Detour (DE)* path and a path without a code that identifies a protected path. Both paths are shown from Ingress Router 4 at Loopback IP address 10.4.4.4 to Egress Router 5 at Loopback IP address 10.5.5.5. The primary path traverses Transit Routers 2 and 6. In the case of the DE path, a reroute sends traffic through Transit Routers 3 and 1.

```
device5# show mpls rsvp session
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
Transit RSVP:     0 session(s)
Egress RSVP:      1 session(s)
To              From             St    Style Lbl_in Lbl_out LSPname
10.5.5.5        10.4.4.4(DE)     Up    SE    3      0       1
10.5.5.5        10.4.4.4         Up    SE    3      0       1
```

The following example displays the output from Egress Router 5 using the **show mpls rsvp session detail** command. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DE path, two PLR and Avoid Node ID pairs are shown labeled [1] and [2]. In both the *Point of Local Repair (PLR)* is at IP Address 10.13.13.1 which is the interface to the primary path on Ingress Router 4. The Avoid Node for pair [1] is IP address 10.13.13.2 on Transit Router 2 and the Avoid Node for pair [2] is IP address 10.18.18.2 on Transit Router 6.

The "Fast Reroute" field indicates that the primary path has been configured for one-to-one backup.

There is no "Explicit path hop count" field for either route because Egress Router 5 is the destination of the path.

```
device5# show mpls rsvp session detail
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
Transit RSVP:     0 session(s)
Egress RSVP:      1 session(s)
```

```
To               From            St    Style Lbl_in Lbl_out LSPname
10.5.5.5         10.4.4.4(DE)    Up    SE    3      0       1
  Time left in seconds (PATH refresh: 18, ttd: 149
                        RESV refresh: 7, ttd: 152)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Detour Rcvd: Number of PLR and Avoid Node ID pair(s): 2
   [1]: PLR: 10.13.13.1  Avoid Node: 10.13.13.2
   [2]: PLR: 10.13.13.1  Avoid Node: 10.18.18.2
  PATH rcvfrom: 10.15.15.1      (e8/2          ) (MD5 OFF)
To               From            St    Styl Lbl_in Lbl_outm LSPname
10.5.5.5         10.4.4.4        Up    SE   3      0        1
  Time left in seconds (PATH refresh: 30, ttd: 152
                        RESV refresh: 7, ttd: 152)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Fast Reroute: one-to-one backup desired
   Setup priority: 7, hold priority: 0
   Bandwidth: 0 kbps, hop limit: 255

PATH rcvfrom: 10.19.19.2      (e8/1) (MD5 OFF)
```

The Transit Router 6 display

The following display examples are from Transit Router 6. Displays are shown for the **show mpls rsvp session** and **show mpls rsvp session detail** commands.

Both displays show two paths from Ingress Router 4 at Loopback IP address 10.4.4.4. to Egress Router 5 at Loopback IP address 10.5.5.5. The (DI) path is an Ingress Detour path, and the path without a code is a protected path. The DI path is the detour path that is taken when Transit Router 6 is unable to use the primary path to Egress Router 5.

```
device6# show mpls rsvp session
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
Transit RSVP:     1 session(s)
To               From            St   Style Lbl_in Lbl_out LSPname
10.5.5.5         10.4.4.4(DI)    Up   SE    1024   1025    1
10.5.5.5         10.4.4.4        Up   SE    1024   3       1
Egress RSVP:      0 session(s)
```

The following example displays the output from Transit Router 6 using the **show mpls rsvp session detail** command. This command provides additional details about the paths described in the output from the **show mpls rsvp session** command.

For the DI path, one PLR and Avoid Node ID pair is shown labeled [1]. In [1], the Point of Local Repair (PLR) is at IP Address 10.19.19.2 which is an interface on Transit Router 6 and the Avoid Node is IP address 0.0.0.0. The "Explicit path hop count" field indicates that there are five hops on this path from this router to the egress to the path at routers with the following IP addresses 10.18.18.1 (Transit Router 2), 10.13.13.1 (Ingress Router 4), 10.10.10.1 (Transit Router 3), 10.11.11.2 (Transit Router 1) and 10.15.15.2 (Egress Router 5).

For the primary path, the "Explicit path hop count" field indicates that there is one hop on this path from this router to the egress to the path to the router at IP address 10.19.19.1 (Egress Router 5). The "Fast Reroute" field indicates that the primary path has been configured for one-to-one backup.

```
device6# show mpls rsvp session detail
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  RP:Repaired Session
Ingress RSVP:     0 session(s)
Transit RSVP:     1 session(s)
To               From              St   Style Lbl_in Lbl_out LSPname
10.5.5.5         10.4.4.4(DI)      Up   SE    1024   1025    1
  Time left in seconds (PATH refresh: 6, ttd: 4293497
                        RESV refresh: 24, ttd: 131)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Explicit path hop count: 5
   10.18.18.1 (S) -> 10.13.13.1 (S) -> 10.10.10.1 (S) -> 10.11.11.2 (S) ->
```

```
    10.15.15.2 (S)
  Received RRO count: 5
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.18.18.1 -> 10.13.13.1 -> 10.10.10.1 ->
    10.11.11.2 -> 10.15.15.2
  Detour Sent: Number of PLR and Avoid Node ID pair(s): 1
   [1]: PLR: 10.19.19.2  Avoid Node: 0.0.0.0
  PATH sentto:  10.18.18.1      (e5/1          ) (MD5 OFF)
  RESV rcvfrom: 10.18.18.1      (e5/1          ) (MD5 OFF)
To               From              St   Style Lbl_in Lbl_out LSPname
10.5.5.5         10.4.4.4          Up   SE    1024   3       1
  Time left in seconds (PATH refresh: 28, ttd: 150
                        RESV refresh: 24, ttd: 128)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 1500
  Fast Reroute: one-to-one backup desired
   Setup priority: 7, hold priority: 0
   Bandwidth: 0 kbps, hop limit: 255
   Detour LSP: UP.  Nexthop (node) protection available.
    Up/Down times: 1, num retries: 0
  Explicit path hop count: 1
   10.19.19.1 (S)
  Received RRO count: 1
   Protection codes: P: Local  N: Node  B: Bandwidth  I: InUse
   10.19.19.1
  PATH rcvfrom: 10.18.18.1      (e5/1          ) (MD5 OFF)
  PATH sentto:  10.19.19.1      (e5/2          ) (MD5 OFF)
  RESV rcvfrom: 10.19.19.1      (e5/2          ) (MD5 OFF)
Egress RSVP:      0 session(s)
```

Configuring MPLS Fast Reroute using one-to-one backup

The **frr** command enables MPLS Fast Reroute using the one-to-one backup on the LSP under whose configuration it is enabled. Options for this command are described in the sections that follow.

Configuring MPLS fast reroute using one-to-one backup

To configure MPLS fast reroute by using the one-to-one backup method for a defined LSP, complete the following steps.

1. Enable the device for configuration.

   ```
   device# configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Configure the LSP.

   ```
   device(config-router-mpls)# lsp frr_tunnel
   ```

   In this example, the specified LSPs name is *frr_tunnel*.

4. Configure the egress router of the LSP.

   ```
   device(config-router-mpls-lsp-frr_tunnel)# to 10.1.1.1
   ```

   In this example, the egress router IP address of the LSP is 10.1.1.1.

5. Configure the primary path.

```
device(config-router-mpls-lsp-frr_tunnel)# primary-path direct_path
```

   In this example, the primary paths name is *direct_path*.

6. Configure the secondary path.

```
device(config-router-mpls-lsp-frr_tunnel)# secondary-path alt_path
```

   In this example, the secondary paths name is *alt_path*.

7. Enable configuration of fast reroute options.

```
device(config-router-mpls-lsp-frr_tunnel)# frr
```

8. Configure bandwidth for the backup LSP.

```
device(config-router-mpls-lsp-frr-tunnel_frr)# bandwidth 100
```

   In this example, the bandwidth is configured to 100 kbits/sec.

9. Configure the number of hops the backup LSP can traverse (from PLR to MP).

```
device(config-router-mpls-lsp-frr-tunnel_frr)# hop-limit 20
```

   In this example, the hop limit is configure a maximum of 20.

In the following example, MPLS Fast Reroute is configured by using the one-to-one backup method for a defined LSP named *frr_tunnel* .

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp frr_tunnel
device(config-router-mpls-lsp-frr_tunnel)# to 10.1.1.1
device(config-router-mpls-lsp-frr_tunnel)# primary-path direct_path
device(config-router-mpls-lsp-frr_tunnel)# secondary-path alt_path
device(config-router-mpls-lsp-frr_tunnel)# frr
device(config-router-mpls-lsp-frr_tunnel-frr)# bandwidth 100
device(config-router-mpls-lsp-frr_tunnel-frr)# hop-limit 20
```

Th frr command enables MPLS Fast Reroute using the one-to-one backup on the LSP under whose configuration it is enabled.

Configuring bandwidth for a MPLS fast reroute

To define a bandwidth constraint for the fast reroute path, complete the following steps.

1. Enable the device for configuration.

```
device# configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the LSP.

```
device(config-router-mpls)# lsp frr_tunnel
```

   In this example, the LSP name is *frr_tunnel* .

4. Enable configuration of fast reroute options.

```
device(config-router-mpls-lsp-frr_tunnel)# frr
```

5. Configure the bandwidth for the backup LSP.

```
device(config-router-mpls-lsp-frr_tunnel-frr)# bandwidth 100
```

   In this example, the bandwidth is configured 100 kbits/sec. for the backup LSP. The acceptable value can be between zero (0) and two (2) Gbps, with 0 being the default value.

The example below defines a bandwidth of 100 kbits/sec. for the fast reroute path.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# lsp frr_tunnel
device(config-router-mpls-lsp-frr_tunnel)# frr
device(config-router-mpls-lsp-frr_tunnel-frr)# bandwidth 100
```

An additional way to add bandwidth to a FRR path is by using the **bandwidth inherit** command.

Configuring priority for a MPLS fast reroute

The user can specify setup and hold priorities for the detour routes within a specified LSP.

These setup and hold priorities for the detour routes are available to any LSP and function the same on standard LSPs as they do on detour LSPs. The priority determines the relative importance of the detour routes during setup or preemption. The priority has two components the setup priority and the hold priority.

When a detour LSP is assigned a higher setup priority, it can preempt any LSP (detour or otherwise) that is already established and has a lower holding priority, causing resources assigned to the lower priority LSP to be diverted to the higher priority LSP. The hold priority specifies how likely an established LSP is to give up its resources to another LSP. To be preempted, an LSP must have a lower hold priority than the preempting LSPs setup priority. In addition, an established LSP can be preempted by a higher priority LSP only when it would allow the higher priority LSP to be established successfully.
To configure the detour routes of LSP *frr_tunnel* with a setup priority of 6 and hold priority of 1, complete the following steps.

1. Enable the device and configure the terminal to the global configuration mode.

```
device>enable
device# configure terminal
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Configure the LSP.

```
device(config-mpls)#lsp frr_tunnel
```

   In this example, the name of the LSP is *frr_tunnel* .

4. Enable configuration of fast reroute options.

```
device(config-mpls-lsp-frr_tunnel)# frr
```

5. Configure the setup and hold priorities.

```
device(config-mpls-lsp-frr_tunnel-frr)# priority 6 1
```

   Possible values for the setup and hold priorities are 0 (highest priority) through 7 (lowest priority). The setup priority must be lower than, or equal to, the configured hold priority on an LSP. By default, the setup priority is 7 and the hold priority is 0.

The following example shows the configuration of the detour routes of LSP *frr_tunnel* with a setup priority of 6 and hold priority of 1.

```
device>enable
device# configure terminal
device(config)# router mpls
device(config-mpls)#lsp frr_tunnel
device(config-mpls-lsp-frr_tunnel)# frr
device(config-mpls-lsp-frr_tunnel-frr)# priority 6 1
```

For additional information regarding the **priority** command, refer to the Fusion Command Line Interface (CLI) Reference Guide.

MPLS OAM

MPLS OAM addresses the requirement to check the data plane functionality of MPLS tunnels.

Use MPLS OAM to detect the data plane health of MPLS tunnels. Specifically, it allows the user to ping and traceroute the MPLS RSVP LSP and LDP tunnels.

Ping MPLS RSVP LSP

Checks the operability of MPLS RSVP label-switched path (LSP) connections.

The **ping mpls rsvp lsp** command can execute the following.

1. Initiate a ping request at the ingress router of an RSVP LSP.
2. Handle the incoming ping request at a transit node of an RSVP LSP (router-alert).
3. Handle an incoming ping request at the egress node of an RSVP LSP.
4. Handle a ping reply at the ingress router of an RSVP LSP.

The following example displays the output of the executed command.

```
device# ping mpls rsvp lsp Test12

Send 5 96-byte MPLS Echo Requests over RSVP LSP Test12, timeout 5000 msec

Type Control-c to abort
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max=0/1/2 ms.
device#
device#
device# ping mpls rsvp session 30.31.32.33 130.130.130.1 1


Send 5 96-byte MPLS Echo Requests for RSVP session 130.130.130.1/1/30.31.32.33, timeout 5000 msec

Type Control-c to abort
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max=0/1/2 ms.
```

The following example is the corresponding **show mpls rsvp session** command output.

```
device# show mpls rsvp session
Codes: DI:Ingress Detour  DT:Transit Detour  DM:Merged Detour
       DE:Egress Detour  BI:Ingress Backup BM: Merged Backup BE:Egress Backup
       RP:Repaired Session BYI: Bypass Ingress

Total Number of such sessions are: 1

Ingress RSVP:     1 session(s)
To              From                St Style Lbl_In  Lbl_Out Out_If    LSPname
130.130.130.1   30.31.32.33         Up FF    -       2050    Eth 1/5   TestDut12
```

Traceroute MPLS RSVP LSP

Traceroute to a remote host for an MPLS label switched path (LSP) signaled by RSVP.

The **traceroute mpls rsvp lsp** command can execute the following.

1. Initiate a traceroute request at ingress router of an RSVP LSP.
2. Handle an incoming traceroute request at a transit node of an RSVP LSP (router-alert).
3. Handle an incoming traceroute request at the egress node of an RSVP LSP.
4. Handle a traceroute reply at the ingress router of an RSVP LSP.

The following example displays the output of the command.

```
device# traceroute mpls rsvp lsp Test12

Trace RSVP LSP Test12, timeout 5000 msec, TTL 1 to 30

Type Control-c to abort
 1  2ms 120.120.120.1 return code 8(Transit)
 2  2ms 130.130.130.1 return code 3(Egress)
device#
device#
device# traceroute mpls rsvp session 30.31.32.33 130.130.130.1 1

Trace RSVP session(130.130.130.1/1/30.31.32.33), timeout 5000 msec, TTL 1 to 30

Type Control-c to abort
 1  2ms 120.120.120.1 return code 8(Transit)
 2  1ms 130.130.130.1 return code 3(Egress)
```

Ping MPLS LDP Tunnel

Checks the operability of the MPLS LDP-signaled label switched path (LSP) connections.

The **ping mpls ldp** command can execute the following.

1. Initiate a ping request at the ingress router of an LDP tunnel.
2. Handle an incoming ping request at a transit node of an LDP tunnel (router-alert).
3. Handle an incoming ping request at the egress node of an LDP tunnel.
4. Handle a ping reply at the ingress router of an LDP tunnel.

The following example displays the output of the command with LDP IP address *130.130.130.1/32*.

```
device# ping mpls ldp 130.130.130.1

Send 5 84-byte MPLS Echo Requests for LDP FEC 130.130.130.1/32, timeout 5000 msec

Type Control-c to abort
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max=0/1/2 ms.
device# ping mpls ldp 130.130.130.1/32

Send 5 84-byte MPLS Echo Requests for LDP FEC 130.130.130.1/32, timeout 5000 msec

Type Control-c to abort
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max=0/0/1 ms.
```

The following example is the corresponding output of the **show mpls ldp tunnel** command.

```
device# show mpls ldp tunnel
 Total number of LDP Tunnels:2
                   Oper      Tunnel     Outbound
To                 State     Intf       Intf
130.130.130.1/32   UP        tnl1       Eth 1/5
120.120.120.1/32   UP        tnl0       Eth 1/5
```

Traceroute MPLS LDP Tunnel

Traceroute to a remote host for a MPLS label switched path (LSP) signaled by the LDP.

1. Initiate traceroute request at ingress router of an LDP tunnel.

```
device# traceroute
```

2. Handle incoming traceroute request at a transit node of an LDP tunnel (router-alert).
3. Handle incoming traceroute request at the egress node of an LDP tunnel.
4. Handle traceroute reply at the ingress router of an LDP tunnel.

```
device# traceroute mpls ldp 130.130.130.1

Trace LDP LSP to 130.130.130.1/32, timeout 5000 msec, TTL 1 to 30

Type Control-c to abort
 1  3ms 120.120.120.1 return code 8(Transit)
 2  1ms 130.130.130.1 return code 3(Egress)

Trace LDP LSP to 130.130.130.1/32, timeout 5000 msec, TTL 1 to 30

Type Control-c to abort
 1  1ms 120.120.120.1 return code 8(Transit)
 2  2ms 130.130.130.1 return code 3(Egress)
```

Auto-Bandwidth

Auto-bandwidth allows for a very efficient use of network-bandwidth.

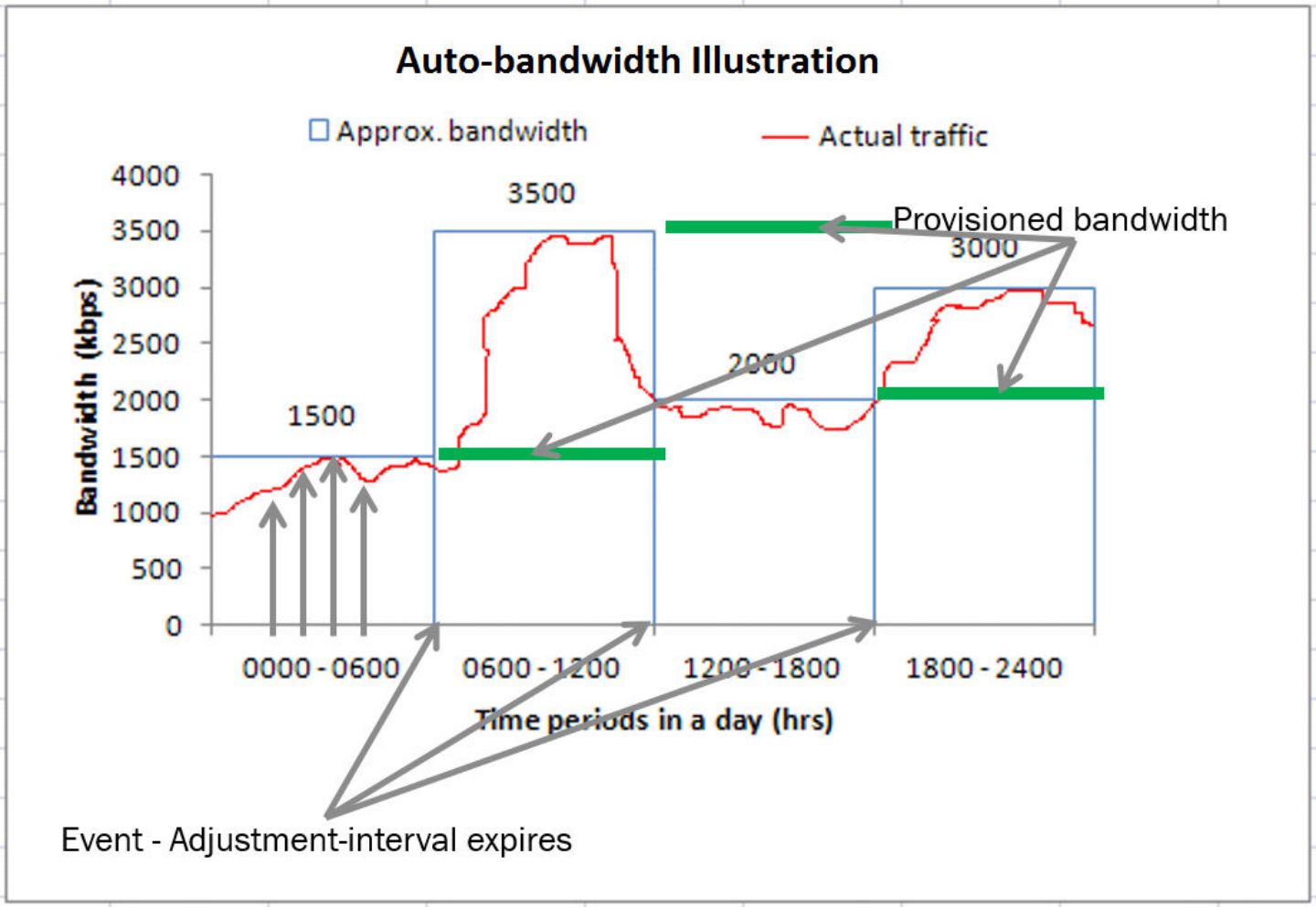
With the auto-bandwidth feature, the traffic rate through an LSP is sampled and the reserved bandwidth of the LSP is automatically changed through a make-before-break mechanism. This is done in order to keep the reserved bandwidth close to the actual traffic rate. It is beneficial to have an optimum bandwidth reservation for an LSP.

Several parameters are available to tune the automatic adjustment behavior. Some examples of parameters are the adjustment-interval, the sample-interval, the adjustment-threshold, and the overflow-limit. The basic auto-bandwidth functionality is that for every adjustment-interval, the predicted bandwidth is calculated as the maximum sampled traffic-rate in the previous adjustment-interval. For example, the adjustment interval is 10 minutes and the sample interval is 1 minute. Every adjustment-period contains 10 samples of traffic rate. Out of these 10 samples, the highest sampled rate is selected as the reserved bandwidth for the next adjustment interval.

Use the make-before-break procedure to change the bandwidth of an LSP without affecting the traffic. This procedure involves the exchange of RSVP control messages. The user may sometimes choose to ignore a bandwidth change when the difference in the current bandwidth and the predicted bandwidth is insignificant. For example, if the current bandwidth is 1 Mbps and the predicted bandwidth is 1.01 Mbps, the delta change is insignificant. To avoid a bandwidth adjustment, in this case, the user can set an adjustment-threshold. The user can configure the adjustment-threshold in terms of percentage in the previous releases. If the user sets the adjustment-threshold to be 10%, the bandwidth of an LSP adjusts only if the different between the current bandwidth and the predicted bandwidth is greater than 10% of the current bandwidth.

The actual traffic rate may not be close to the predicted rate always, given the very nature of prediction. In order to adapt to a higher traffic rate, it may be required to increase the reserved bandwidth of the LSP sooner than waiting for the adjustment-interval to expire. To achieve this, there is a parameter called the overflow-limit. If the overflow-limit is set to 3, and 3 consecutive samples of actual traffic-rate are found to exceed the current bandwidth by an amount greater than the configured threshold, a premature adjustment is triggered, setting the bandwidth to the maximum of the samples obtained so far in the adjustment-interval.

FIGURE 26 Basic auto-bandwidth functionality



Auto-bandwidth components

Auto-bandwidth consists of the following components:

- Configurable table-based absolute adjustment-threshold
- Underflow-limit configuration and functionality
- Record and display history of sampled values for an LSP

Configurable table-based absolute adjustment-threshold

One way to achieve a threshold is to define a table which can give the absolute threshold based on the current traffic rate. Below is an example of how a typical threshold table looks. An additional column is added for illustrating that how the percentage threshold varies as the current bandwidth increases.

TABLE 8 Configurable table-based absolute adjustment-threshold

Range of actual traffic rate	Threshold	Percentage threshold range
0-1000 kbps	2000 kbps	?? - 200%
1000 kbps to 10 Mbps	3000 kbps	300% - 30%
10 Mbps to 100 Mbps	5000 kbps	50% - 5%
100 Mbps to 1Gbps	7000 kbps	7% - 4%

Note that the absolute threshold values increase with the actual-traffic rate but the percentage threshold decreases. This way, the user can make sure that for low traffic-rate LSPs, insignificant bandwidth changes are ignored. This saves costly make-before-break procedures and provides a scalability benefit over the current uniform percentage based method. This is particularly beneficial in cases where a router is having LSPs with a wide range of actual-traffic rates.

Pros of the threshold-table based method:

- Useful when there are LSPs with wide range of actual traffic rate
- Huge scalability benefit. Avoid insignificant bandwidth adjustments

It is important that the range of values and the corresponding thresholds are chosen carefully.

The percentage-based threshold method and table-based threshold methods co-exist. There is another option to configure if an LSP is using the percentage-based threshold or the table-based threshold. Note that there is a single global table only to be used system-wide by all LSPs. An LSP is allowed to chose from either the threshold-table or the LSP level percentage threshold configured. This flag behaves in the same way as the other LSP level auto-bandwidth parameters. This flag also allows configuration on an auto-bandwidth template. It follows the same inheritance mechanism as other parameters. This threshold is valid for both overflow and underflow determination.

Underflow-limit

This parameter can be configured along the same lines as overflow-limit. This parameter is configurable on an auto-bandwidth template as well as an LSP. The same inheritance rules as other parameters will apply.

Suppose the underflow-limit is set to 10. If 10 consecutive samples of the LSP traffic rate are found to be lesser than the LSP bandwidth by an amount more than the threshold, a premature adjustment is triggered setting the LSP bandwidth to the maximum of these 10 consecutive samples. **The sampled-rate chosen as the new bandwidth for the LSP will be the maximum of those 10 samples that triggered the underflow.** Note that if adjustment-threshold is configured to use the **autobw-threshold-table**, the threshold from the auto-bandwidth table will be used. Unlike overflow-limit, **the number of underflow counts is not reset after adjustment-interval expiry.** This means that out of the 10 samples that triggered the adjustment, six may be the beginning samples of the current adjustment period while the remaining four may be the last samples of the previous adjustment period.

Sample-history

The user is given the option to record all the events related to auto-bandwidth of an LSP using the CLI command "**sample-recording enable/disable**".

The samples obtained in an adjustment-interval can be displayed whenever needed with the show command "**show mpls lsp autobw-samples**". The history also contains relevant auto-bandwidth events such as adjustments.

The user is given the freedom to clear or delete the auto-bandwidth samples at any point of time using the CLI command "**clear mpls auto-bandwidth-samples**". The auto-bandwidth history is deleted only in the cases when LSP is itself deleted or when the user clears or deletes the samples manually. Clearing of the auto-bandwidth samples by user is recorded in the LSP history.

Configuring the threshold-table for adjustment threshold

Use the **autobw-threshold-table** command at the MPLS configuration level to begin configuration of the table. This command leads to a new parser mode **config-mpls-autobw-threshold-table**. Under this mode, commands to remove and add a row in the table is provided.

To set the adjustment-threshold table, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable the auto-bandwidth threshold table.

```
device(config-router-mpls)# autobw-threshold-table
```

4. Configure the auto-bandwidth threshold table.

```
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling 10 threshold 2000
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling 1000 threshold 3000
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling 10000 threshold 5000
```

5. NOTE
   The user is also required to set the default scenario. The case from the last threshold configured to maximum bandwidth value. If the user does not set this values, the threshold value for the maximum (previous) range is used.

   Configure the default. In this example, the default is configured to 10%.

```
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling max threshold percentage 10
-or-
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling max threshold 10000
```

The following example combines the steps above to configure the adjustment-threshold table.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# autobw-threshold-table
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling 10 threshold 2000
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling 1000 threshold 3000
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling 10000 threshold 5000
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling max threshold percentage 10
-or-
device(config-mpls-autobw-threshold-table)# bandwidth-ceiling max threshold 10000
```

Removing threshold entries

To remove one of the threshold entries, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable the auto-bandwidth threshold table.

```
device(config-router-mpls)# autobw-threshold-table
```

4. Remove the threshold entry by using the **no** form of the command.

```
device(config-mpls-autobw-threshold-table)#no bandwidth-ceiling 1000 threshold 3000
```

The following example combines the steps above to remove one of the threshold entries.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# autobw-threshold-table
device(config-mpls-autobw-threshold-table)#no bandwidth-ceiling 1000 threshold 3000
```

Clearing the threshold table

To clear the threshold table, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Clear the threshold table using the **no** form of the command.

```
device(config-mpls)# no autobw-threshold-table
```

   The table is cleared with this command. If the LSP or template is set to use the global table, the adjustment-threshold considered would be zero (0) (default).

The following example combines the steps above to clear the threshold table.

```
device>configure
device(config)# router mpls
device(config-mpls)# no autobw-threshold-table
```

Configuring an auto-bandwidth template to use the global table for adjustment threshold.

To configure an auto-bandwidth template to use the global table for adjustment threshold, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable the auto-bandwidth template. In this example, template 1 is selected.

```
device(config-mpls)# autobw-template template1
```

4. Enable the global table.

```
device(config-mpls-autobw-template-template1)# adjustment-threshold use-threshold-table
```

The following example combines the steps above to configure auto-bandwidth template *1* to use the global table for adjustment threshold.

```
device>configure
device(config)# router mpls
device(config-mpls)# autobw-template template1
device(config-mpls-autobw-template-template1)# adjustment-threshold use-threshold-table
```

Configuring an LSP to use the global table for adjustment threshold

To configure an LSP to use the global table for adjustment threshold, complete the following steps.

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Select and enable LSP. In this example LSP1 is selected.

   ```
   device(config-mpls)# lsp lsp1
   ```

4. Enable auto-bandwidth.

   ```
   device(config-mpls-lsp-lsp1)# autobw
   ```

5. Configure the adjustment threshold to use the threshold table.

   ```
   device(config-mpls-lsp-lsp1-autobw)# adjustment-threshold use-threshold-table
   ```

the following example combines the steps above to to configure an *LSP1* to use the global table for adjustment threshold.

```
device>configure
device(config)# router mpls
device(config-mpls)# lsp lsp1
device(config-mpls-lsp-lsp1)# autobw
device(config-mpls-lsp-lsp1-autobw)# adjustment-threshold use-threshold-table
```

Configure the LSP or template to use percentage values.

To configure the LSP to use the percentage values, complete the following steps

1. Enable the device for configuration.

   ```
   device>configure
   ```

2. Enable the MPLS router.

   ```
   device(config)# router mpls
   ```

3. Select and enable LSP. In this example LSP1 is selected.

   ```
   device(config-mpls)# lsp lsp1
   ```

4. Enable auto-bandwidth.

   ```
   device(config-mpls-lsp-lsp1)# autobw
   ```

5. Configure the adjustment threshold to use the percentage values.
   The user issues the same command with the **no** option. The LSP now uses the percentage threshold.

```
device(config-mpls-lsp-lsp1-autobw)# no adjustment-threshold use-threshold-table
```

The following example combines the steps above to configure the LSP to use the percentage values.

```
device(config)# router mpls
device(config)# router mpls
device(config-mpls)# lsp lsp1
device(config-mpls-lsp-lsp1)# autobw
device(config-mpls-lsp-lsp1-autobw)# no adjustment-threshold use-threshold-table
```

Configuring the underflow limit

The user can configure the underflow limit at the LSP level as well as in the auto-bandwidth template. The inheritance rules applied are the same as other auto-bandwidth parameters. The default value of underflow-limit is zero (0).

To configure the underflow-limit in an auto-bandwidth template, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable the auto-bandwidth template and select template. In this example, *template1* is selected.

```
device(config-router-mpls)# autobw-template template1
```

4. Configure the underflow limit. In this example, the underflow limit is configured to *10*.

```
device(config-router-mpls-autobw-template-template1)# underflow-limit 10
```

The following example combines the steps above to set the underflow-limit in an auto-bandwidth template.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# autobw-template template1
device(config-router-mpls-autobw-template-template1)# underflow-limit 10
```

Configuring the underflow-limit for an individual LSP.

To configure the underflow-limit for an individual LSP, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable LSP configuration ans select target LSP. In this example, *LSP1* is selected.

```
device(config-router-mpls)# lsp lsp1
```

4. Enable auto-bandwidth.

```
device(config-mpls-lsp-lsp1)# autobw
```

5. Enable and configure the underflow-limit. In this example, the underflow-limit is configured to 10.

```
device(config-router-mpls-lsp-lsp1-autobw)# underflow-limit 10
```

In the following example, the steps above are combined to configure the underflow-limit for individual *LSP1* to *10* .

```
device>configure
device(config)# router mpls
device(config-router-mpls)# lsp lsp1
device(config-mpls-lsp-lsp1)# autobw
device(config-router-mpls-lsp-lsp1-autobw)# underflow-limit 10
```

Clearing the underflow-limit configuration

To clear the underflow-limit configuration, use the same command as above with the **no** option. The underflow-limit is set back to the default value of zero (0).

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable LSP configuration ans select target LSP. In this example, *LSP1* is selected.

```
device(config-router-mpls)# lsp lsp1
```

4. Enable auto-bandwidth.

```
device(config-mpls-lsp-lsp1)# autobw
```

5. Enable and configure the underflow-limit. In this example, the underflow-limit is configured to 10.

```
device(config-router-mpls-lsp-lsp1-autobw)# no underflow-limit 10
```

In the following example, the steps above are combined to clear the underflow-limit configuration and set the configuration back to the default value of zero (*0*).

```
device>configure
device(config)# router mpls
device(config-router-mpls)# lsp lsp1
device(config-mpls-lsp-lsp1)# autobw
device(config-router-mpls-lsp-lsp1-autobw)# no underflow-limit 10
```

Displaying the sample-history

The user can configure an LSP or template to enable or disable the sample recording indicating that the history of samples be recorded for that LSP or template. By default, the sample recording is **disabled** for an LSP. An option is provided in the **show mpls lsp** commands to display the auto-bandwidth history for the auto-bandwidth LSPs.

The samples are recorded for every adjustment-interval irrespective of whether an adjustment was done or not. For example, if the adjustment did not happen because of threshold is not being crossed or the mode is configured as **monitor-only** or auto-bandwidth is

disabled globally or at the LSP level or when the LSP is itself disabled or the auto-bandwidth statistics are cleared, the sample history for that adjustment interval is displayed with the proper messages.

When the user wants to record the sample history for an LSP, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable LSP configuration and select the target LSP. In this example, the selected LSP is *LSP1*.

```
device(config-router-mpls)# lsp lsp1
```

4. Enable auto-bandwidth.

```
device(config-router-mpls-lsp-lsp1)# autobw
```

5. Enable sample-recording.

```
device(config-router-mpls-lsp-lsp1-autobw)# sample-recording enable
```

The following example combines the steps above to enable sample-recording on *LSP1*.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# lsp lsp1
device(config-router-mpls-lsp-lsp1)# autobw
device(config-router-mpls-lsp-lsp1-autobw)# sample-recording enable
```

Stop recording sample history

To stop recording sample history for an LSP, complete the following steps.

1. Enable the device for configuration.

```
device>configure
```

2. Enable the MPLS router.

```
device(config)# router mpls
```

3. Enable LSP configuration and select the target LSP. In this example, the selected LSP is *LSP1*.

```
device(config-router-mpls)# lsp lsp1
```

4. Enable auto-bandwidth.

```
device(config-router-mpls-lsp-lsp1)# autobw
```

5. Disable sample-recording.

```
device(config-router-mpls-lsp-lsp1-autobw)# sample-recording disable
```

The following example combines the steps above to disable recorded sample history on *LSP1*.

```
device>configure
device(config)# router mpls
device(config-router-mpls)# lsp lsp1
device(config-router-mpls-lsp-lsp1)# autobw
device(config-router-mpls-lsp-lsp1-autobw)# sample-recording disable
```

To display the sample record, use the following command.

```
device# show mpls lsp autobw-samples
```

To clear all the recorded sample history, use the following command.

```
device# clear mpls auto-bandwidth-sample-history
```

The user may also wish to record the sample-history for certain period of time. The user is able to turn off and on the sample recording to capture the rate for those periods. In order completely delete the sample-history and turn off recording, the user is required to first turn off the sample-recording followed by clearing the sample history.

Special circumstance behaviors

The user toggles the option to use the global table

The user can change this configuration on the fly. In this case, the new value of the adjustment-threshold is used when the next sample is obtained. The user is not expected to use this option too frequently. If an adjustment is already pending based on the previous adjustment-threshold, the adjustment is made.

The user sets a lower absolute threshold value for a higher bandwidth value

Typically, the values of the threshold increase with the bandwidth. This means that as the ceiling value of bandwidth increases, the threshold value also increases. There is no restriction on this behavior. If the user sets a lower adjustment-threshold value for a higher range of bandwidth, a warning message displays. Consider the case below. If the user configure it this way, the configuration is accepted but a warning message displays.

Range	Threshold
0-10 kbps	2000 kbps
10 - 1000 kbps	3000 kbps
1000 - 10000 kbps	**2000 kbps << The threshold is less than the previous threshold**
10000 kbps +	10000 kbps

The user changes the global table values

Does not trigger any immediate event. The new values are considered from the next sample onwards. When an adjustment is already pending based on the previous adjustment-threshold, the adjustment is done.

The user changes the underflow-limit in the template or at the LSP level.

Does not trigger any immediate event. The new values are considered from the next sample onwards. When an adjustment is already pending based on the previous underflow-limit value, the adjustment is done.

The user disables the sample-recording for an LSP or a template.

When the user disables sample-recording for an LSP or a template, the sample-recording is stopped and no further samples are recorded. Sentinels are added to indicate that the sample-recording was stopped due to user configuration.

The user clears the sample-recording history for an LSP

When the user clears the sample history for an LSP, the recorded sample history is deleted completely from the memory. To completely remove the recording behavior, the user must un-configure the sample recording followed by clearing the sample-record history.

Label Distribution Protocol

LDP overview

When used to create LSP tunnels, LDP allows a set of destination IP prefixes (known as a Forwarding Equivalence Class or FEC) to be associated with an LSP.

Each Link Switch Router (LSR) establishes a peer relationship with the neighboring LDP-enabled routers and exchanges label mapping information. This label mapping information is stored in an LDP database on each LSR. When an LSR determines that one of the peers is the next-hop for a FEC, the LSR uses the label mapping information from the peer to set up an LSP that is associated with the FEC.

The devices advertise their loopback addresses to their LDP peers as a 32-bit prefix-type FEC. When an LSR installs a label for a FEC, it also creates an MPLS tunnel route, which is then made available to routing applications. This allows each router to potentially be an ingress LER for an LSP whose destination is the device's loopback address.

The result of an LDP configuration is a full mesh of LSPs in an MPLS network, with each LDP-enabled router a potential ingress, transit, or egress LSR, depending on the destination.

The system supports LDP for the configuration of non-traffic-engineered tunnel LSPs in an MPLS network. LDP is described in *RFC 5036*.

The LDP label space ID has a default value of zero which improves interoperability with routers from other vendors.

LDP terminology

Before implementing LDP, familiarize yourself with the following key terms and definitions.

TABLE 9 LDP terminology

Apply Current Route	Compare the current route with the received label mappings and install any downstream mappings, as appropriate.
Current Route	Current next hop info for a FEC. The current route may not be applied immediately to the current label mapping as a result of LWD at ingress.
Downstream mapping (DM)	Represents the label mapping received from a downstream peer for a FEC.

TABLE 9 LDP terminology (continued)

FEC	Forwarding Equivalency Class. Each FEC is a destination IP address for an LDP tunnel.
LDP	Label Distribution Protocol.
Label mapping	LDP message that indicates the label to be used for an FEC from the peer.
LSP	Label Switched Path.
LWD	Label Withdrawal Delay.
Route event	Update from the routing table to LDP.
Upstream Mapping	Represents the label mapping sent to an upstream peer for a FEC.

Configuring LDP on an interface

For an LDP session between routers, you must configure LDP on an interface to allow the device to advertise its loopback interface to the peers.

To use LDP, configure a loopback address with a 32-bit mask on the LSR. The first loopback address configured on the device is used in its LDP identifier. When the loopback address used in the LDP identifier is removed, all LDP functions on the LSR are shut down. LDP sessions between the LSR and its peers are terminated, and LDP-created tunnels are removed. When other loopback interfaces are configured on the device, the lowest-numbered loopback address is used as a new LDP identifier. LDP sessions and tunnels are set up using this new LDP identifier.

Configure LDP on the same set of interfaces that IGP routing protocols such as OSPF and IS-IS are enabled.

To configure LDP on an interface, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.
2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Enable MPLS on an interface.

```
device(config-router-mpls)# mpls-interface ethernet 1/2
```

   For example, MPLS is enabled on the interface at Ethernet 1/2.

4. Enable LDP on the interface.

```
device(config-router-mpls-if-eth-1/2-ldp-params)# ldp-enable
```

5. Return to privileged EXEC mode.

```
device(config-router-mpls-if-eth-1/2-ldp-params)#)# Ctrl-z
```

6. Verify the configuration of the interface.

```
device# show mpls ldp interface
             Label-space    Nbr      Hello       Next
Interface    ID             Count    Interval    Hello
e1/2         0              1        5           0 sec
```

The following example shows the previous steps to configure LDP on an interface.

```
device# configure
device(config)# router mpls
device(config-router-mpls)# mpls-interface ethernet 1/2
device(config-router-mpls-if-eth-1/2)# ldp-params
device(config-router-mpls-if-eth-1/2-ldp-params)# ldp-enable
```

Configuring the LDP session keepalive interval

You can configure the LDP session keepalive interval and the keepalive timeout value is derived as the product of the keepalive interval times the keepalive interval count.

If the **ka-timeout** command is configured on your device, you must explicitly remove its configuration with the **no ka-timeout** command before you change the keepalive interval. For example:

```
device(config-router-mpls-ldp)# ka-timeout 40
%Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
device(config-router-mpls-ldp)# ka-interval 11
%Error: ka-timeout needs to be unconfigured before ka-interval is configured
device(config-router-mpls-ldp)# no ka-timeout
%Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
device(config-router-mpls-ldp)# ka-interval 11
%Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
```

To configure the keepalive intervals, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

   ```
   device# configure terminal
   ```

2. Enable MPLS on the device.

   ```
   device(config)# router mpls
   ```

3. Access LDP configuration mode.

   ```
   device(config-router-mpls)# ldp
   ```

4. Set the keepalive time interval at which the session keepalive message is sent when no other LDP protocol message is sent to the LDP peer.

   ```
   device(config-router-mpls-ldp)# ka-interval 10
   %Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
   ```

   In this example, the keepalive interval is 10 seconds. The possible values are 1 through 65535. A warning is displayed whenever the **ka-interval** value is changed.

5. Configure the number of keepalive intervals after which the session is terminated when no session keepalive or other LDP protocol message is received from the LDP peer.

   ```
   device(config-router-mpls-ldp)# ka-int-count 3
   %Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
   ```

   In this example, the number of intervals is set to 10.

6. Verify the keepalive interval configuration.

```
device # show mpls ldp
Label Distribution Protocol version 1 (deleting it will stop LDP)
 LSR ID: 10.122.122.122, using Loopback 1
 Hello interval: Link 5 sec, Targeted 15 sec
 Hold time value sent in Hellos: Link 15 sec, Targeted 45 sec
 Keepalive interval: 10 sec, Hold time multiple: 3 intervals
 Keepalive timeout: 30 sec
 Load sharing: 8
 Tunnel metric: 0
 FEC used for auto discovered peers: current 129, configured 129
 End of LIB: Disabled, Notification time: 60000 ms, tx label silence time: 1000 ms
 Rx label silence time: 1000 ms
 Graceful restart: disabled
  Reconnect time: 0 seconds, Max peer reconnect time: 120 seconds
  Recovery time: 0 seconds, Max peer recovery time: 120 seconds
  Forwarding state holding timer: not running
 Label Withdrawal delay: 60 seconds (Default)
```

   When the keepalive interval is configured, the keepalive timeout value displays as the product of keepalive interval * keepalive interval count. In this example, the keepalive timeout set to 30.

The following example is the configuration of the previous steps.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# ka-interval 10
device(config-router-mpls-ldp)# ka-int-count 3
```

Configuring the LDP session keepalive timeout

After an LDP session is established, an LSR maintains the integrity of the session by sending keepalive messages. The keepalive timer for each peer session resets whenever it receives any LDP protocol message or a keepalive message on that session. When the keepalive timer expires, LDP concludes that the TCP connection is bad or the peer is dead and terminates the session.

The **ka-interval** and **ka-timeout** command configurations are mutually exclusive. You can only have one configured at a time. You must explicitly remove the keepalive interval before changing the keepalive timeout. For example:

```
device(config-router-mpls-ldp)# ka-interval 11
%Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
device(config-router-mpls-ldp)# ka-timeout 40
%Error: ka-interval needs to be unconfigured before ka-timeout is configured
device(config-router-mpls-ldp)# no ka-interval
%Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
device(config-router-mpls-ldp)# ka-timeout 40
%Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
```

To configure the keepalive timeout and number of intervals, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

4. Set the keepalive time interval at which the session is terminated when the keepalive or LDP protocol message is not received.

```
device(config-router-mpls-ldp)# ka-timeout 180
%Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
```

   In this example, the time after which the session is terminated when the keepalive or LDP protocol message is not received is set to 180 seconds. The possible values are 1 through 65535. A warning is displayed whenever the **ka-timeout** value is changed.

5. Configure the number of keepalive intervals after which the session is terminated when no session keepalive or other LDP protocol message is received from the LDP peer.

```
device(config-router-mpls-ldp)# ka-int-count 10
%Warn: Please clear LDP sessions for the new KA parameter value to take effect on existing sessions
```

   In this example, the number of intervals is set to 10.

6. Verify the keepalive timeout configuration.

```
device # show mpls ldp
Label Distribution Protocol version 1 (deleting it will stop LDP)
 LSR ID: 10.122.122.122, using Loopback 1
 Hello interval: Link 5 sec, Targeted 15 sec
 Hold time value sent in Hellos: Link 15 sec, Targeted 45 sec
 Keepalive interval: 18 sec, Hold time multiple: 10 intervals
 Keepalive timeout: 180 sec
 Load sharing: 8
 Tunnel metric: 0
 FEC used for auto discovered peers: current 129, configured 129
 End of LIB: Disabled, Notification time: 60000 ms, tx label silence time: 1000 ms
 Rx label silence time: 1000 ms
 Graceful restart: disabled
  Reconnect time: 0 seconds, Max peer reconnect time: 120 seconds
  Recovery time: 0 seconds, Max peer recovery time: 120 seconds
  Forwarding state holding timer: not running
 Label Withdrawal delay: 60 seconds (Default)
```

   When the keepalive timeout value is configured, the keepalive interval displays as keepalive timeout divided by the keepalive interval count (ka-timeout/ka-in-count).

The following example is the configuration of the previous steps.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# ka-timeout 180
device(config-router-mpls-ldp)# ka-int-count 10
```

LDP Hello interval and Hello Hold timeout timers

The LDP Hello interval and Hello Hold Timeout timers are used to establish Hello Adjacency between peers. The Hello interval is the time period between which the LSR sends out Hello messages and the Hello Hold timeout is the amount of time that the sending LSR maintains its record of Hellos from the receiving LSR without receipt of another Hello message.

The Hello interval and Hello Hold Timeout timer values can be obtained from the global default values or configured globally on a router. The Hello Hold timeout timer value can also be configured through an interface. When configuring these values the following constraints must be followed:

- The Hello Interval value must be less than 32767.
- The Hello Hold Timeout value must be less than 65535.
- The Hello Hold Timeout value must be greater than or equal to 2 times the Hello Interval value.

The values can be set that determine the values used on the configured router and values sent to adjacent peers for their configuration as follows:

- Setting the LDP Hello interval values
- Setting the LDP Hold time sent to adjacent LSRs
- Determining the LDP Hold time on an MPLS interface

LDP Hello interval

The LDP hello interval controls how often the device sends out LDP Hello messages. Hello messages are used to maintain LDP adjacencies between the device and its LDP peers.

You can set the interval for LDP Link Hello messages (LDP Hello messages multicast to all routers on the sub-net), as well as for LDP Targeted Hello messages (LDP Hello messages unicast to a specific address, such as a VLL peer):

- For targeted LDP adjacencies—The LDP Hello Interval can only be set globally. When a Hello Interval is not set for targeted LDP adjacencies, then the global default value is used.
- For link LDP adjacencies—The LDP Hello Interval can be set globally which applies to all LDP interfaces or on a per-interface basis. The LDP Hello Interval values in Link LDP adjacencies are determined by the following procedure in the order.
  1. When the Hello Interval is set per-interface, this value is used.
  2. When the Hello Interval is not set per-interface, then the value set for LDPs globally is used.
  3. When the Hello Interval is not set either globally or per-interface, the global default value is used.

     When a Hello adjacency already exists, the adjacency remains up and any new configured interval takes effect upon the expiration of the current Hello Interval timer. Consequently, the next and subsequent hello messages are sent at the new interval.

LDP Hello Hold time

The LDP hold time specifies how long the device or MPLS interface waits for its LDP peers to send a Hello message.

When the device does not receive a Hello message within this time, the LDP session with the peer can be terminated. The device includes the hold time in the Hello messages it sends out to its LDP peers. The LDP Hold time sent in Hello messages to adjacent LSRs can be configured globally for either Link or Targeted LDP sessions.

For an MPLS interface, how long it waits for its LDP peers to send a Hello message differs for a targeted LDP session and a link LDP session, as follows:

- For targeted LDP sessions—The value received in Hello messages from its peers determines the time that the device waits for its LDP peers to send a Hello message. When the timeout value received from a peer is zero, the Hold time is set to the default period of 45 seconds.
- For link LDP sessions —In this case, the wait time is determined by any one of the following criteria.
  - When the Hello hold time is set per-interface, that value is used.
  - When the Hello hold time is not set per-interface, the hold time in the received message is used.
  - When the Hello hold time in the received message is zero (0), the default value of 15 seconds is used.

Changing the LDP Hello interval

You can change the default setting for the LDP Hello message interval globally for link or target LDP sessions. You can also configure the interval on an interface for link LDP sessions.

To change the global link and target intervals for LDP Hello messages, and configure a link interval for an interface, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

4. Change the interval for the link sessions.

```
device(config-router-mpls-ldp)# hello-interval-link 10
```

   In this example, the hello interval set to 10. The default value is 5. You can enter an integer from 1 through 32767.

   NOTE
   The Hello interval for link sessions can be overridden on a per-interface basis.

5. Change the interval for the target sessions.

```
device(config-router-mpls-ldp)# hello-interval-target 20
```

   In this example, the hello interval set to 20. The default value is 15. You can enter an integer from 1 through 32767. The change takes effect immediately.

   NOTE
   This value can only be set globally for all targeted LDP sessions on the router. Per-interface configuration is only available for Link LDP sessions.

6. Access MPLS mode.

```
device(config-router-mpls-ldp)# exit
```

7. Enable MPLS on an interface.

```
device(config-router-mpls)# mpls-interface ethernet 1/2
```

   For example, MPLS is enabled on interface on Ethernet port 1/2.

8. Configure LDP parameters on the interface.

```
device(config-router-mpls-interface-1/2)# ldp-params
```

9. Change the interval for link sessions on the interface.

```
device(config-router-mpls-interface-1/2-ldp-params)# hello-interval 15
```

   In this example, the hello interval set on this interface for LDP Link Hello messages is set to 15 seconds. You can enter an integer from 1 through 65535. No default value exists for this parameter. However, when no value is set for this parameter, it defaults to the global LDP Hello interval. When you set a hello interval on the interface, it overrides the global LDP Hello interval.

The following example shows the previous steps to configure the LDP Hello interval.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# hello-interval-link 10
device(config-router-mpls-ldp)# hello-interval-target 20
device(config-router-mpls-ldp)# exit
device(config-router-mpls)# mpls-interface ethernet 1/2
device(config-router-mpls-interface-1/2)# ldp-params
device(config-router-mpls-interface-1/2-ldp-params)# hello-interval 15
```

Changing the LDP hold time sent to adjacent LSRs

You can change the default settings for the LDP Hold Time sent in Hello messages to adjacent LSRs globally or on an interface for either link or targeted LDP sessions.

To change the hold time included in LDP Hello messages, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

4. Change the hold time for link sessions.

```
device(config-router-mpls-ldp)# hello-timeout-link 10
```

   In this example, the hello hold time set to 10. The default value is 15. You can enter an integer from 2 through 65335.

   NOTE
   The Hello hold time for link sessions can be overridden on a per-interface basis.

5. Change the hold time for target sessions.

```
device(config-router-mpls-ldp)# hello-timeout-target 20
```

   In this example, the hello interval set to 20. The default value is 45. You can enter an integer from 2 through 65335.

   The new time takes effect immediately and goes in the next Hello message sent. This hold time applies to only the hold time that the device sends to its peers; it does not affect the hold time the device uses to time out those peers. The latter is determined from the hold time that peers send to the device.

6. Access MPLS mode.

```
device(config-router-mpls-ldp)# exit
```

7. Enable MPLS on an interface.

```
device(config-router-mpls)# mpls-interface ethernet 1/2
```

   For example, MPLS is enabled on interface Ethernet port 1/2.

8. Configure LDP parameters on the interface.

```
device(config-router-mpls-interface-1/2)# ldp-params
```

9. Change the hold time for link sessions on the interface.

```
device(config-router-mpls-interface-1/2-ldp-params)# hello-timeout 30
```

   In this example, the hello hold time on this interface for LDP Link Hello messages is set to 30 seconds. The minimum value that you can configure is 2 times the value set for the Hello interval.

The following example shows the previous steps to configure the LDP Hello hold time.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# hello-timeout-link 10
device(config-router-mpls-ldp)# hello-timeout-target 20
device(config-router-mpls-ldp)# exit
device(config-router-mpls)# mpls-interface ethernet 1/2
device(config-router-mpls-interface-1/2)# ldp-params
device(config-router-mpls-interface-1/2-ldp-params)# hello-timeout 30
```

Configuring LDP message authentication

To protect against spoofed TCP segments in a connection stream, Extreme devices allow configuration of an authentication key on a per LDP session basis.

The LDP session can be to an adjacent peer (basic discovery) or to the targeted peer (extended discovery). You must configure both sides of an LDP peer link.

The software supports LDP authentication based upon the TCP MD5 signature option specified in *RFC 2385*. This RFC defines a new TCP option for carrying an MD5 digest in a TCP segment.

To configure LDP message authentication, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

4. Configure an authentication key on an LDP session.

```
device(config-router-mpls-ldp)# session 10.10.10.3 key early
```

   In this example, the IP address of the LDP peer for authentication is 10.10.10.3. The encrypted text string between the peers is early and it must be configured on both peers.

The following example shows the previous steps to configure LDP message authentication.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# session 10.10.10.3 key early
```

When you display the configuration, the key is encrypted.

```
show running-config router mpls ldp
router mpls
 ldp
  session 10.10.10.3
   key $9$+DysjCDsYS2ba9EW4i6SPA==
  !
 !
!
```

Resetting LDP neighbors

You can terminate and re-establish an MPLS LDP neighbor session when at least one LDP Hello adjacency exists with the peer. When the LDP session terminates, the database associated with the LDP session is also cleared.

The LDP sessions are automatically reestablished when at least one "hello" adjacency exists with the neighbor, and LDP configuration remains unchanged.

Enter the **clear mpls ldp neighbor** command to terminate an LDP session.

```
device# clear mpls ldp neighbor 10.234.123.64
```

In this example, both the link and targeted LDP sessions on neighbor 10.234.123.64 are terminated. When the **all** option is specified instead of a peer address, all LDP sessions on the device are reset.

When the session re-establishes, the session-specific information is re-learned from its peer:

- LDP downstream and upstream label database displayed by the **show mpls ldp database** command.
- LDP label switched path displayed by the **show mpls ldp path** command.
- LDP peer displayed by the **show mpls ldp peer** command
- LDP created MPLS tunnels displayed by the **show mpls ldp tunnel** command.
- LDP FECs learned from the resetting neighbor sessions displayed by the **show mpls ldp fec** command. FECs are not cleared immediately but are marked that no LDP session exists.

Validating LDP session reset

You can check the LDP session specific parameters to validate that a session has been successfully reset.

- The LDP session state transitions from Operational to Nonexistent upon clearing it. However, the states may quickly transition. In this case, the **show mpls ldp session** command shows the Up time, and that it was reset to zero upon clearing the session.
- The LDP session specific database is cleaned upon resetting the LDP session.
- The TCP port number on the active end of the LDP session may have been changed once the LDP session comes up after the reset; the TCP port number before and after the reset may be different. Use the **show mpls ldp session** command to view the TCP port number.
- Syslog logs the event of a LDP session going down and then coming back up as a result of resetting the LDP session. Use the command **show log** to view the syslog events.

The following command shows two LDP sessions with neighbor 10.234.123.64.

```
device# show mpls ldp session
 Number of link LDP sessions: 1
 Number of Operational link LDP sessions: 1
 Number of targeted LDP sessions: 0
 Number of Operational targeted LDP sessions: 0
Peer LDP ID              State          Adj Used  My Role   Max Hold  Time Left
10.234.123.64            Operational    Link      Passive   36        33
```

The following command clears both the link and targeted LDP session with neighbor 10.234.123.64.

```
device# clear mpls ldp neighbor 10.234.123.64
device#
device# show mpls ldp session
Peer LDP ID              State          My Role   Max Hold  Time Left
10.234.123.64            Operational    Passive   36        33
```

This command shows that after waiting for roughly 20 seconds (depends on the hello or keepalive timer periodicity), both the LDP sessions are reestablished.

```
device# show mpls ldp session
Peer LDP ID              State          My Role   Max Hold  Time Left
10.234.123.64            Operational    Passive   36        33
```

LDP route injection

By default, LDP advertises all /32 prefixes that are learned from all the loopback interfaces to all other LDP peers. LDP route injection enables LDP to advertise other prefixes that are learned by IGP. When you enable route injection, it references prefix lists for permitted and denied prefixes. When IGP learns these prefixes, the device injects them into LDP and advertises the corresponding labels to the LDP peers.

Considerations when using LDP route injection

- You can change the LDP route injection filter without deleting a previously configured one. The change automatically applies and triggers LDP route re-injections.
- Any change to a referenced prefix list automatically applies to LDP route injection filtering and triggers LDP route re-injection.
- When no LDP route injection filter is configured, by default, LDP acquires all local loopback addresses.
- When the prefix list referenced by the LDP route injection filter is not configured, it is an implicit deny. All local routes are denied.
- The LDP route injection filter is only applied on local route injection. Learned remote binding is not filtered.

Configuring LDP route injection

Configure LPD route injection to allow the Extreme device to inject routes into LDP by referencing a prefix list and advertise the FEC to the LDP peers. By default, LDP advertises all /32 prefixes that are learned from the loopback interfaces to all other LDP peers.

To configure LDP route injection, perform the following steps:

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Configure the prefix list to filter the route.

```
device(config)# ip prefix-list list1 permit 10.2.2.2/32
```

   In this example, a filter is configured to inject route 10.2.2.2/32.

3. Enter router MPLS configuration mode.

```
device(config)# router mpls
```

4. Enter LPD configuration mode.

```
device(config-router-mpls)# ldp
```

5. Enter the **advertise-fec** command to configure LDP route injection.

```
device(config-router-mpls-ldp)# advertise-fec list1
```

   LDP route injection is configured to inject the routes by referencing the list1 prefix list into LDP and advertise the FEC to the LDP peers.

6. Access privileged EXEC mode.

```
device(config-router-mpls-ldp)# Ctrl-z
```

7. Verify the LDP route injection configuration.

```
device# show running-config router mpls ldp
router mpls
 ldp
  ....
  advertise-fec list1
 !
!
```

8. Verify that the LDP route injection has been injected into the LDP label information database.

```
device# show mpls ldp database
Session 10.3.3.3:0 - 10.5.5.2:0
 Downstream label database:
   Label     Prefix           State
 Upstream label database:
   Label     Prefix
   3         10.2.2.2/32
```

The following configuration is an example of the previous steps.

```
device(config)# ip prefix-list list1 permit 10.2.2.2/32
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# advertise-fec list1
```

LDP route injection example

The following example describes LDP route injection of routes 10.2.2.2/32 and 10.5.5.2/32 into the LDP label information database.

By default, the LDP label information database only contains labels learned for IP addresses of loopback interfaces, as demonstrated in this example, where only prefixes 10.3.3.3/32 and 10.5.5.5/32 are displayed by the **show mpls ldp database** command.

```
device# show mpls ldp database
Session 10.3.3.3:0 - 10.5.5.2:0
 Downstream label database:
   Label     Prefix          State
   1024      10.3.3.3/32     Retained
```

```
 Upstream label database:
   Label     Prefix
   3         10.3.3.3/32
   3         10.5.5.5/32
```

A filter is configured to inject route 10.2.2.2/32.

```
device# configure terminal
device(config)#ip prefix-list list1 permit 10.2.2.2/32
device(config)# router mpls
device(config-mpls)# ldp
device(config-router-mpls-ldp)# advertise-fec list1
device(config-router-mpls-ldp)# Ctrl-z
```

The **show mpls ldp database** command displays that route 10.2.2.2/32 has been injected into the LDP Label information database.

```
device# show mpls ldp database
Session 10.3.3.3:0 - 10.5.5.2:0
 Downstream label database:
   Label     Prefix          State
 Upstream label database:
   Label     Prefix
   3            10.2.2.2/32
```

A second filter is configured to inject route 10.5.5.2/32.

```
device# configure terminal
device(config)# ip prefix-list list1 permit 10.5.5.2/32
device(config)# exit
```

The **show mpls ldp database** command displays that route 10.5.5.2/32 has been injected into the LDP label information database.

```
device# show mpls ldp database
Session 10.3.3.3:0 - 10.5.5.2:0
 Downstream label database:
   Label     Prefix          State
 Upstream label database:
   Label     Prefix
   3            10.2.2.2/32
   3            10.5.5.2/32
```

LDP inbound-FEC filtering

MPLS LDP inbound-FEC filtering filters inbound label bindings on a MPLS router. You can control the amount of memory and CPU processing involved in installing and advertising label bindings not used for forwarding.

MPLS LDP inbound-FEC filtering also serves as a tool to avoid DOS attack. By creating a prefix-list, and specifying prefixes label mappings, the forwarding plane accepts and installs the label bindings. The prefix-list is applied to an individual LDP session or globally to all the LDP sessions.

Configuration considerations for LDP inbound-FEC filtering

- The FECs filtered by the LDP inbound-FEC filter do not install in the forwarding plane or advertise to the upstream neighbors. The FEC remains in the retained state.
- The LDP inbound-FEC filter are changed directly without deleting the one previously configured. The change automatically applies and triggers the filtering of inbound FECs.
- Changes to a referenced prefix-list automatically applies to LDP inbound-FEC filtering. This triggers filtering by way of the new configuration, filtering any existing FECs which violate the filter.

- To allow multiple route filter updates, the device waits for default 10 seconds before notifying the application of the filter change. The time for notification is configurable.
- When the LDP inbound-FEC filter is not configured, LDP does not filter any inbound FECs.
- By default, when the prefix-list referenced by the LDP inbound-FEC filter has no configuration, it is an implicit deny. All inbound FECs are filtered out and retained. The behavior is the same when the prefix list is deleted after setting it in the inbound FEC filter configuration. This behavior is consistent with other protocols which use device filters and also with the use of the **advertise-fec** command for LDP route injection.
- Inbound FEC filtering is applicable only for Layer 3 FECs and not for VC FECs. Inbound FEC filtering is not applicable for Layer 2 VPNs.

Configuring LDP inbound FEC filtering

Configure LDP inbound-FEC filtering to filter inbound label bindings on a MPLS router.

To enable LDP inbound FEC filtering,

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Configure the prefix list to be referenced by the LDP inbound-FEC filter.

```
device(config)# ip prefix-list list-abc permit 10.20.20.0/24
```

   In this example, the list-abc prefix list allows the 10.20.20.0/24 route address.

3. Enable MPLS on the device.

```
device(config)# router mpls
```

4. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

5. Configure the LDP inbound FEC filter.

```
device(config-router-mpls-ldp)# filter-fec-in list-abc
```

   In this example, LDP accepts inbound FEC 10.20.20.0/24 through the list-abc prefix list and filter out all others FECs.

   NOTE
   When the prefix list referenced by the LDP inbound-FEC filter is configured or changed, all the existing in-bound FECs and received later are subject to the changed prefix list. There is a configurable delay between changing the prefix list and the changed prefix list taking effect on LDP FEC-filter configuration.

6. Verify the inbound FEC-filter configuration.

```
device(config-router-mpls-ldp)# do show running-config router mpls ldp
router mpls
 ldp
  ....
  filter-fec-in list-abc
 !
!
```

The following example shows the previous steps to configure the LDP inbound-FEC filter.

```
device# configure terminal
device(config)# ip prefix-list list-abc permit 10.20.20.0/24
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# filter-fec-in list-abc
```

Displaying inbound FEC filter information

In addition to the Inbound FEC-filter configuration, the **show mpls ldp** command displays the following inbound FEC-filter information.

- Prefix list and filtered information
- Filtered information
- Label database for the sessions when there is at least one mapping which is filtered due to the FEC-filter action
- Number of FECs from the peer which are filtered due to the inbound FEC filter configuration

Displaying the prefix list and filtered information

To display the prefix list and filtered information, use the **show mpls ldp fec prefix prefix-filter** command. In the following example, the downstream mapping for the inbound FECs is filtered by the inbound-FEC filter prefix list and is displayed as a retained state. The output displays that it is retained due to the inbound FEC-filter action.

```
device# show mpls ldp fec prefix prefix-filter 172.16.8.0/24
FEC_CB: 0x2cd83d78, idx: 4, type: 2, pend_notif: None, fec_definition:22080000
 State: current, Ingr: Yes, Egr: No, UM Dist. done: Yes
 Prefix: 172.16.8.0/24
 next_hop: 10.55.55.14, out_if: e3/16

 Downstream mappings:
 Local LDP ID     Peer LDP ID      Label    State           CB
 10.44.44.44:0    10.14.14.14:0    1024     Retained (f)    0x2cd3b610(-1)
```

Displaying the filtered information

To display the filtered information, use the **show mpls ldp fec prefix filtered** command.

```
device# show mpls ldp fec prefix filtered
Total number of prefix FECs: 1607
Total number of prefix FECs installed: 1505
Total number of prefix FECs filtered(in/out): 0/0
Total number of prefix FECs with LWD timer running: 0

Destination         State    Out-intf         Next-hop        Ingress Egress Filtered LWD
10.44.44.44/32        current  --                 --              No      Yes
10.66.66.66/32        current  Eth 4/2          10.1.1.2        Yes     No
10.14.14.14/32        current  Eth 3/16         10.55.55.14     Yes     No
```

Displaying the label database for the filtered mapped sessions

To display the label database for the sessions when there is at least one mapping which is filtered due to the FEC-filter filter action, use the **show mpls ldp database** command. In the following example, the inbound FECs filtered by the inbound-FEC filter prefix list are displayed as a retained state.

```
device# show mpls ldp database
Session 10.44.44.44:0 - 10.14.14.14:0
 Downstream label database:
   Label      Prefix            State
   3          10.14.14.14/32    Installed
   1024          172.16.8.0/24     Retained(filtered)
```

```
    1025          172.16.16.0/24    Retained(filtered)
    1026          172.16.32.0/24    Retained(filtered)
    1027          172.16.64.0/24    Retained(filtered)
    1028          172.16.8.0/28     Retained(filtered)
    1029          172.16.8.16/28    Retained(filtered)
    1030          172.16.8.32/28    Retained(filtered)
    1031          172.16.8.64/28    Retained(filtered)
Upstream label database: Label    Prefix
       3        10.44.44.44/32
       1024     10.66.66.66/32
     Session 10.44.44.44:0 - 10.66.66.66:0
Downstream label database:
       Label    Prefix          State
       3        10.66.66.66/32  Installed
Upstream label database: Label  Prefix
       3        10.44.44.44/32
       1025     10.14.14.14/32
```

Displaying the number of FECs from the filtered peer

To display the number of FECs from the peer which are filtered due to the inbound FEC filter configuration, use the **show mpls ldp session detail** command.

```
device# show mpls ldp session detail
Number of link LDP sessions: 1
Number of Operational link LDP sessions: 1
Number of targeted LDP sessions: 0
Number of Operational targeted LDP sessions: 0

Peer LDP ID: 10.66.66.66:0, Local LDP ID: 10.44.44.44:0, State: Operational
  Adj: Link, Role: Passive, Next keepalive: 2 sec, Hold time left: 32 sec
  Keepalive interval: 6 sec, Max hold time: 36 sec
  Configured keepalive timeout: 36 sec
  Peer proposed keepalive timeout: 36 sec
  Up time: 21 sec
  TCP connection: 10.44.44.44:646--10.66.66.66:9075, State: ESTABLISHED
  Neighboring interfaces: Eth 4/2
  Next-hop addresses received from the peer:7
  10.1.1.2 10.66.66.66 10.168.1.1 10.168.1.2 10.168.1.3 10.168.1.4    10.168.1.5
  IGP Sync:
    Unrecognized Notification Capability: Local: Off, Remote: Off
    Local State: In-sync, RemoteState: -
    Rx label silence time: 1000 ms, Timer not running
  Graceful restart: enabled
  Number of FECs Received from peer: 18
  Number of FECs installed from peer: 5
  Number of FECs filtered from peer(in/out): 1/0
```

FEC filtering configuration example

The following examples use the FEC filtering parameter.

Consider three MPLS router system devices with an ID 10.66 with the transit device between them.

FIGURE 27 Inbound FEC filtering example

The following configuration configures the prefix list to allow all /32 addresses, the prefix list to allow 172.16.0.0/16 ge 24 le 24, prefix list to allow172.16.0.0/16 ge 24 le 28, and a prefix list to allow all of the previous FECs.

```
device(config)# ip prefix filter172_24 permit 172.16.09.0/16 ge 24 le 24
device(config)# ip prefix filter172_24 permit 172.16.09.0/16 ge 24 le 24
device(config)# ip prefix-list filter172_28 permit 172.16.0.0/16 ge 24 le 28
device(config)# ip prefix-list filterAll permit 0.0.0.0/0 ge 32
device(config)# ip prefix-list filterAll permit 172.16.0.0/16 ge 24 le 28
```

Verify the configuration by using the **show mpls ldp** command.

```
device(config)# exit
device# show mpls ldp
Label Distribution Protocol version 1
  LSR ID: 10.44.44.44, using Loopback 1 (deleting stops LDP)
  Hello interval: Link 5 sec, Targeted 15 sec
  Hold time value sent in Hellos: Link 15 sec, Targeted 45 sec
  Keepalive interval: 6 sec, Hold time multiple: 6 intervals
  Load sharing: 1
  Tunnel metric: 0
  FEC used for auto discovered peers: current 129, configured 129
  Graceful restart: disabled
device(config)# do show mpls ldp database
Session 10.44.44.44:0 - 10.14.14.14:0
 Downstream label database:
   Label        Prefix            State
   3            10.14.14.14/32    Installed
   1024         172.16.8.0/24     Installed
   1025         172.16.16.0/24    Installed
   1026         172.16.32.0/24    Installed
   1027         172.16.64.0/24    Installed
   1028         172.16.8.0/28     Installed
   1029         172.16.8.16/28    Installed
   1030         172.16.8.32/28    Installed
   1031         172.16.8.64/28    Installed
 Upstream label database:
   Label        Prefix
   3            10.44.44.44/32
   1033         10.66.66.66/32
Session 10.44.44.44:0 - 10.66.66.66:0
 Downstream label database:
   Label        Prefix            State
   3            10.66.66.66/32    Installed
 Upstream label database:
   Label        Prefix
   3            10.44.44.44/32
   1024         172.16.8.0/24
   1025         172.16.16.0/24
   1026         172.16.32.0/24
   1027         172.16.64.0/24
   1028         172.16.8.0/28
   1029         172.16.8.16/28
   1030         172.16.8.32/28
   1031         172.16.8.64/28
   1032         10.14.14.14/32
```

LDP outbound FEC filtering

LDP outbound FEC filtering allows LDP to perform outbound filtering for label advertisement. It gives you the ability to control which FECs can be advertised and to which LDP neighbors. It also reduces the number of labels distributed to neighbors and the number of messages exchanged with peers. Through this feature, LDP scalability and convergence, security, and performance are improved.

LDP performs a hop-by-hop or dynamic path setup in an MPLS network by assigning and distributing labels to routes learned from the underlying IGP routing protocols. By default, LDP distributes all FECs that are learned locally or from LDP neighbors to all other LDP neighbors. When this behavior is not desired, you can configure LDP to perform outbound filtering FEC filtering.

Outbound filtering is achieved by creating a prefix list that specify prefixes whose label mappings can be distributed. The prefix list is applied to an individual LDP neighbor, or globally to all the LDP neighbors. The FECs permitted by the prefix list are accordingly distributed to the specified LDP neighbor or to all LDP neighbors.

Prerequisites

MPLS and LDP protocols must be enabled on the router to use this feature.

Configuring global LDP outbound FEC filtering

Configure global LDP outbound FEC filtering to allow or prevent the advertisement of FECs to all neighbors.

MPLS and LDP protocol must be enabled on the router to use LDP outbound FEC filtering.

To enable global LDP outbound FEC filtering, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

   ```
   device# configure terminal
   ```

2. Configure the prefix list to be referenced by the LDP outbound FEC filter to deny an address.

   ```
   device(config)# ip prefix-list list-out deny 10.40.40.0/24
   ```

   In this example, the list-out prefix list prevents the 10.40.40.0/24 route address.

3. Configure the prefix list to allow all other FECs to all neighbors.

   ```
   device(config)# ip prefix-list list-out permit permit 0.0.0.0/0 ge 32
   ```

   In this example, the list-out prefix list allows the default route address.

4. Enable MPLS on the device.

   ```
   device(config)# router mpls
   ```

5. Access LDP configuration mode.

   ```
   device(config-router-mpls)# ldp
   ```

6. Configure the LDP outbound FEC filter.

   ```
   device(config-router-mpls-ldp)# filter-fec-out list-out
   ```

   In this example, LDP prevents advertisement of FEC 10.40.40.0/24 through the list-out prefix list and allows all others FECs to all neighbors.

   NOTE

   When the prefix list referenced by the LDP outbound FEC filter is configured or changed, all the existing outbound FECs and received later are subject to the changed prefix list. There is a configurable delay between changing the prefix list and the changed prefix list taking effect on LDP FEC filter configuration.

7. Verify the outbound FEC-filter configuration.

```
device(config-router-mpls-ldp)# do show running-config router mpls ldp
router mpls
 ldp
  ....
  filter-fec-out list-out
 !
!
```

The following example shows the previous steps to configure the LDP outbound FEC filter.

```
device# configure terminal
device(config)# ip prefix-list list-out deny 10.40.40.0/24
device(config)# ip prefix-list list-out permit 0.0.0.0/0 ge 32
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# filter-fec-out list-out
```

Configuring neighbor-based LDP outbound FEC filtering

Configure neighbor-based LDP outbound FEC filtering to allow or prevent the advertisement of FECs to a specified neighbor.

MPLS and LDP protocol must be enabled on the router to use LDP outbound FEC filtering.

To enable neighbor-based LDP outbound FEC filtering, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Configure the prefix list to be referenced by the LDP outbound FEC filter to deny an address.

```
device(config)# ip prefix-list list-out deny 10.40.40.0/24
```

   In this example, the list-out prefix list prevents the 10.40.40.0/24 route address.

3. Configure the prefix list to allow all other FECs.

```
device(config)# ip prefix-list list-out permit permit 0.0.0.0/0 ge 32
```

   In this example, the list-out prefix list allows the default route address.

4. Enable MPLS on the device.

```
device(config)# router mpls
```

5. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

6. Configure the LDP outbound FEC filter for a specific neighbor.

```
device(config-router-mpls-ldp)# session 10.12.12.12 filter-fec-out list-out
```

   In this example, LDP prevents advertisement of FEC 10.40.40.0/24 through the list-out prefix list and allows all others FECs to neighbor 10.12.12.12.

> NOTE
> When the prefix list referenced by the LDP outbound FEC filter is configured or changed, all the existing outbound FECs and received later are subject to the changed prefix list. There is a configurable delay between changing the prefix list and the changed prefix list taking effect on LDP FEC filter configuration.

7. Verify the LDP FEC filter configuration.

```
device(config-router-mpls-ldp)# do show running-config router mpls ldp
router mpls
 ldp
  ....
  session 10.12.12.12 filter-fec-out list-out
  session 1.1.1.1
   filter-fec-out l2
   key $9$PVUIbBsn+r80zIlBCUpdHw==
```

The **show mpls ldp session detail** command displays the number of FECs from the peer which are filtered due to the outbound FEC filter configuration.

The following example shows the previous steps to configure the LDP outbound FEC filter.

```
device# configure terminal
device(config)# ip prefix-list list-out deny 10.40.40.0/24
device(config)# ip prefix-list list-out permit 0.0.0.0/0 ge 32
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# session 10.12.12.12 filter-fec-out list-out
device(config-router-mpls-ldp)# session 1.1.1.1 filter-fec-out l2 key $9$PVUIbBsn+r80zIlBCUpdHw==
```

Label withdrawal delay timer

The label withdrawal delay timer allows you to configure a delay when sending a label withdraw message for a FEC to a neighbor.

When an LDP session fails, the label associated with a FEC is withdrawn from all upstream peers. In addition, if the IGP adjusts the route for the FEC such that the current neighbor is no longer the next hop for the FEC, then the associated FEC label is withdrawn from all upstream peers.

The label withdrawal delay timer allows you to configure a delay to allow the IGP and LDP to converge after these events. The delay helps avoid sending the label withdraw message to the upstream peers. For example, after a link failure, instead of immediately sending a label withdraw to all upstream peers for the FEC, the delay allows the IGP to install a route which may match another existing downstream session. Label withdrawal from all upstream peers can be avoided if the FEC achieves a downstream label mapping which is consistent with the IGP routing table.

If the timer expires, the FEC label is withdrawn from all upstream peers.

Session down event

When a session becomes inactive, each DM for the session is deleted. This may cause LW to be sent upstream if the last installed DM is removed for a FEC. The behavior is as follows:

- If a DM is a candidate to be removed, the system evaluates whether or not its FEC is a candidate for delaying the LW upstream, based on whether there are other installed DMs
- If there are no other installed DMs, the system proceeds as follows:
  - A timer is started on the FEC for the LW delay period.
  - The DM remains installed until either a new DM becomes installed for the FEC, the route for the FEC is deleted, or the FEC timer expires.
- If there are other installed DMs, then the DM is removed as per normal.

Route update event

When a route update for the FEC occurs, the installed DM may transition to retained if the next hop address for the FEC has changed. The behavior is as follows:

- The route update may result in a FEC with the following criteria:
  - The LW delay timer is not already running for the FEC.
  - The FEC would reference only retained DMs after processing the route update. Note there must be at least one DM.
- If these criteria are met, the system proceeds as follows:
  - A timer is started on the FEC for the LW delay period.
  - The current installed DM remains installed.
  - The current route information is updated. The route is not applied until there is a LM which matches the route, or the LW delay timer expires.
- If these criteria are not met, then LWD is not activated and normal procedures for the FEC are done.
- The existing behavior for link update or route update is not changed if the LWD is not activated for the FEC.

Label withdrawal delay at ingress

Label withdrawal delay at ingress provides a timer that defers the reaction of LDP to session down and routing events to allow the network to stabilize. By default, LWD at ingress is enabled along with the LWD transit automatically and does not require configuration.

Label Withdraw Delay improves network convergence for an LDP network by avoiding sending Label Withdraw protocol messages after specific events in the network. It is helpful for performance at ingress nodes that have not advertised a label to an upstream peer.

LWD at ingress has the following benefits:

- Reduces outage time for traffic originating from ingress tunnels.
- Reduces message generation within internal MPLS modules and line cards.

This feature starts the LWD timer even for purely ingress FECs. The behavior is the same as for the FEC with upstream labels other than procedures for cleaning up the upstream labels on LWD expiration.

The use cases for LWD remain intact with the adjustment that the timer is started even when there are no upstream labels.

Label withdrawal delay and LDP graceful restart

Label withdrawal delay behaves differently when LDP graceful restart is executed by the helper node or the restarting node.

Helper node

When graceful restart is executed by the helper node, graceful restart helper procedures are initiated when a session to a peer goes down. During the session down processing, the label mappings exchanged with a peer are preserved while the peer is reconnecting. If the peer session is reconnected within a configurable time limit then the label mappings previously exchanged with the peer are refreshed with new label mappings. Any mappings that are not refreshed are released.

When both label withdrawal delay and GR are enabled, the label withdrawal delay timer is not initiated when the session goes down because the session is considered to be in a special restarting state and not actually down. If the session is not re-established within the reconnect time for GR then the session is considered to be down and the label withdrawal delay timer may be started for any FECs which meet the criteria for label withdrawal delay. If the session is re-established within the reconnect time for GR then the label withdrawal delay timer is not started.

Individual FECs may experience some transition as the label mappings from the peer are refreshed. For example, if a label mapping is not refreshed during the restart window, then that label mapping will be removed. This will affect the FEC, especially if it was associated with an installed downstream mapping. If an alternate route exists, the FEC will re-converge on the alternate route much earlier and again the label withdrawal delay timer is not started for the FEC

Restarting node

When graceful restart is executed by the restarting node, the GR restarting procedures are executed by a node which is starting the LDP process and has retained the forwarding state for LDP connections from an earlier instance of LDP. The restarting procedures are executed during a management module switchover. During the forwarding state hold period, connections in the forwarding state are marked stale until they are refreshed by an additional label mapping. Stale connections are removed after this period.

This scenario does not affect label withdrawal delay since it involves a complete restart of the LDP control plane. In other words, there are no FECs or sessions to apply the label withdrawal delay timer to at restart. Other procedures for label withdrawal delay as described previously may occur in the same way during the forwarding state hold period.

Label withdrawal delay and LDP-IGP synchronization

LDP-IGP synchronization aims to prevent traffic loss due to the introduction of a new link into the network. Appropriate label mappings for a FEC may not be available for a period of time after the route for the FEC has been established on the new link.

When LDP-IGP synchronization is enabled, the IGP metric for the new link is temporarily advertised at a maximum value to force traffic to use an alternate route, if one is available. After all label mappings are received on the link, the IGP metric is adjusted on the link to the normal value and route updates may occur as the cost of the link has been reduced.

When both label withdrawal delay and LDP-IGP synchronization are enabled, the label withdrawal delay timer will not be started if there are alternate routes for the FEC. For example, the following sequence of events is possible:

1. A FEC has an installed downstream mapping over link 1.
2. Link 2 is introduced to the network. The IGP metric for link 2 is advertised at the maximum value and there is no route update for any FEC with an already established route.
3. Label mappings are received from the new peer and a new retained downstream mapping is established for the FEC.
4. When all label mappings for the session on link 2 have been received, the IGP metric is adjusted to the normal value. By default, the LDP-IGP synchronization hold down time is disabled and IGP will wait until LDP gives an in-sync indication for the link before advertising it with the normal metric. If the LDP-IGP synchronization hold-down time is enabled and label mappings are not received from the new peer within the configured synchronization hold-down time period, then the label withdrawal delay timer will start for the FEC. In this case, it is the time at which the label withdrawal delay timer starts that is affected. Instead of starting almost immediately after the new link becomes operational, it is delayed by the configured time to allow for LDP-IGP synchronization.
5. This may result in a route change at the FEC. The FEC may install the downstream mapping associated with link 2 and transition the downstream mapping associated with link 1 to retained.

Configuring the label withdrawal delay timer

The label withdrawal delay timer delays sending a label withdrawal message for a FEC to a neighbor. This feature is enabled by default with a delay of 60 seconds.

To configure the label withdrawal delay timer, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

4. Configure the delay timer.

```
device(config-router-mpls-ldp)# label-withdrawal-delay 100
```

   In this example, the timer is set to 100 seconds. The default value is 60 seconds.

   If you set the timer to 0 (zero), the label withdrawal delay feature is disabled for subsequent events. Any FEC which has already started the label withdrawal delay timer continues to run the timer and to delay sending its label withdrawal messages upstream.

The following example shows the previous configuration steps.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# label-withdrawal-delay 100
```

LDP ECMP for transit LSR

LDP Equal-Cost Multi-Path (ECMP) for transit LSR provides ECMP support for transit routers on an LDP LSP.

> NOTE
> The SLX 9850 does not support ECMP at an ingress LER.

ECMP programming for LDP transit LSP creates a set of ECMP paths on the forwarding plane at any transit router. LDP LSPs transit traffic is load balanced using programmed ECMP. The number of ECMP paths that are used depends on the number of eligible paths that are available, and the maximum number of LDP ECMP paths that you configure.

The Routing Information Base (RIB) Manager controls the number of available paths sent to LDP which is limited by IP load sharing. LDP also enables its own load sharing limit. The lesser of the two load sharing limits form the maximum number of ECMP paths that can be programmed on forwarding plane.

When new ECMP paths are added, or existing paths are deleted from a set of eligible ECMP paths, MPLS forwarding decides when these changes lead to a different set of paths to be used for LDP LSP, ingress tunnel, or transit LSP.

When a different set of paths are used, updates are sent to the forwarding plane. MPLS only sends an update to the forwarding plane when there is a change to the set of programmed paths. MPLS always sends the complete set of ECMP paths to the forwarding plane. When you change the load sharing configuration, updates are also sent to the forwarding plane. FEC updates are only generated when the new load sharing value is different from the set of ECMP paths programmed in the forwarding plane.

NOTE
LDP ECMP is not supported at the ingress router.

The ingress LDP LSP can be different from the transit LSP for the same FEC.

MPLS OAM support for LDP ECMP

MPLS Operations, Administration, and Maintenance (OAM) support for traceroute at any transit router returns the list of labels used at that transit router. However, traceroute is not able to exercise all ECMP paths. The forwarding plane selects one ECMP path to forward OAM packets. All traversed labels that were returned at each transit router are displayed at the Extreme router originating the traceroute.

Changing the maximum number of LDP ECMP paths

The number of LDP ECMP paths for transit LSR depends on the number of eligible paths that are available, and the maximum number of LDP ECMP paths that you can configure. By default, the maximum number of LDP ECMP paths is one.

To change the maximum number of LDP ECMP paths, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

4. Change the maximum number of LDP ECMP paths.

```
device(config-router-mpls-ldp)# load-sharing 4
```

   In this example, the maximum number of path is set to 4. The default value is 1. You can enter an integer from 1 to 16.

5. Verify the load sharing configuration.

```
device# show mpls ldp
Label Distribution Protocol version 1
LSR ID: 10.125.125.1, using Loopback 1(deleting it stops LDP)
Hello interval: Link 5 sec, Targeted 15 sec
Hold time value sent in Hellos: Link 15 sec, Targeted 45 sec
Keepalive interval: 6 sec, Hold time multiple: 6 intervals
load-sharing 4
```

   NOTE
   The load sharing configuration is displayed only when the configured value is different from the default value.

The following example shows the previous steps to change the maximum number of ECMP paths.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# load-sharing 4
```

MPLS LDP-IGP synchronization

MPLS LDP-IGP synchronization provides a means to synchronize LDP and IGPs to minimize MPLS packet loss.

MPLS LDP-IGP synchronization also provides the following benefits:

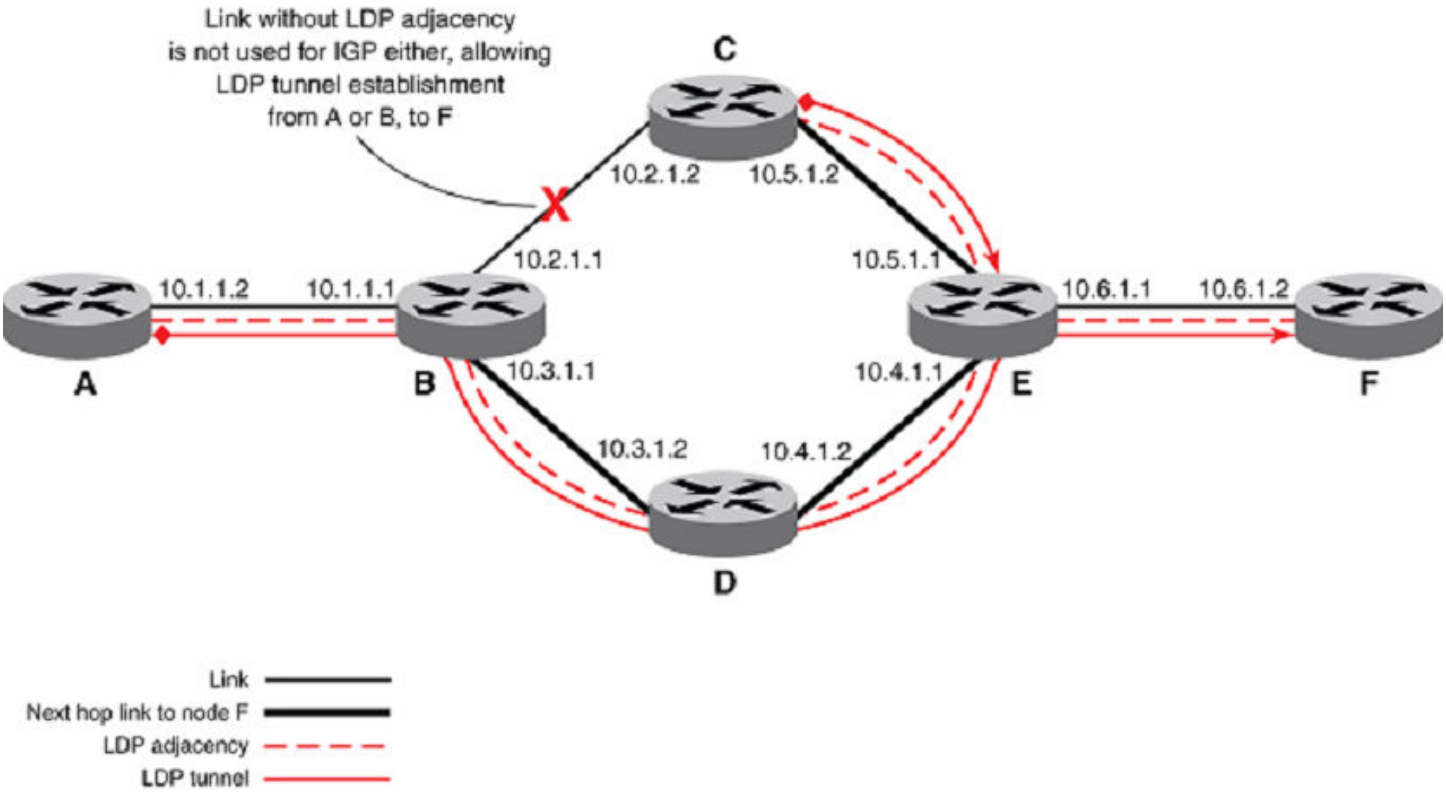
- Provides a means to disable LDP-IGP synchronization on interfaces that you do not want enabled
- Allows you to globally enable LDP-IGP synchronization on each interface associated with an IGP Open Shortest Path First (OSPF) or IS-IS process. OSPF and IS-IS each operate independently.

MPLS LDP-IGP synchronization may be enabled on an interface or globally. LDP determines convergence (receipt of all labels) for a link through one of two methods:

- Receive Label silence mechanism
- End Of Lib mechanism (*RFC 5919*)

The following figure provides an example of LDP-IGP synchronization.

FIGURE 28 Example with LDP-IGP synchronization



When the LDP peer is not reachable, the IGP establishes the adjacency to enable the LDP session to be established. When an IGP adjacency is established on a link but LDP-IGP Synchronization is not yet achieved or is lost, the IGP advertises the maximum metric (max-metric) on the link.

Configuration considerations

- Supports only point-to-point interfaces but not tunnel interfaces

- On IS-IS, wide metric-style is required
- When enabled on IS-IS, the feature applies to both level-1 and level-2 metrics
- Affects IPv4 metrics only

LDP-IGP synchronization hold-down time

The LDP-IGP synchronization hold-down time in router OSPF and the router IS-IS modes is the interval which the IGP must advertise the maximum IP metric, while waiting for an update from LDP. By default, the hold-down time is disabled. IGP waits until LDP gives an In Sync indication for the link before it advertised the normal metric.

The hold down interval starts whenever the IGP initially is enabled with LDP-IGP synchronization. It is also started whenever LDP updates the IGP with an update indicating the interface status is not-in-sync. When the hold down time expires, the IGP resumes advertising the normal metric for the link.

You can configure the hold down time. When you initially configure the hold down time, the router starts the hold-down-timer on every not-in-sync interface at the time.

When you delete the hold down configuration, the router stops the hold-down timer on every interface that is running the hold-down timer running as if there is no hold down time configured. As a result, these interfaces have an infinite hold down time. For the not-in-sync interfaces with an expired hold-down time, IGP continues to advertise the normal metric.

Configuring LDP-IGP synchronization

You can configure LDP-IGP synchronization with an IS-IS or OSPF process globally or on an interface.

Configuring MPLS LDP-IGP synchronization and hold time globally

MPLS LDP-IGP synchronization is disabled by default. You can globally enable MPLS LDP-IGP synchronization with IS-IS and OSPF and configure the hold time setting.

To globally configure MPLS LDP-IGP synchronization with IS-IS or OSPF, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable OSPF and enter OSPF router configuration mode.

```
device(config)# router ospf
```

3. Configure LDP-IGP synchronization.

```
device(config-router-ospf-vrf-default-vrf)# ldp-sync
```

4. Set the hold time to advertise the maximum IP metric while waiting for an update from LDP.

```
device(config-router-ospf-vrf-default-vrf)# ldp-sync hold-down 100
```

   In this example, the global hold time is set to 100 seconds. The default setting is 30. The range is from 1 through 65535.

5. Enter global configuration mode.

```
device(config-router-ospf-vrf-default-vrf)# exit
```

6. Enter IS-IS router configuration mode.

```
device(config)# router isis
```

7. Enable IPv4 address-family configuration mode.

```
device(config-isis-router)# address-family ipv4 unicast
```

8. Configure the generation and acceptance of new-style TLVs.

```
device(config-router-isis-ipv4u)# metric-style wide
```

On IS-IS, wide metric-style is required.

9. Configure LDP-IGP synchronization.

```
device(config-router-isis-ipv4u)# ldp-sync
```

10. Set the hold time to advertise the maximum IP metric while waiting for an update from LDP.

```
device(config-router-isis-ipv4u)# ldp-sync hold-down 100
```

In this example, the global hold time is set to 100 seconds. The range is from 1 through 65535.

11. Verify the LDP-IGP synchronization configuration with OSPF.

```
device# show ip ospf
OSPF Version                Version 2
Router Id                   10.1.1.2
ASBR Status                 No
ABR Status    No    (0) Redistribute Ext Routes from
Initial SPF schedule delay   0         (msecs)
Minimum hold time for SPFs   0         (msecs)
Maximum hold time for SPFs   0         (msecs)
External LSA Counter         0
External LSA Checksum Sum    00000000
Originate New LSA Counter    9
Rx New LSA Counter           6
External LSA Limit           174762
Database Overflow Interval   0
Database Overflow State :    NOT OVERFLOWED
RFC 1583 Compatibility :    Enabled
Slow neighbor Flap-Action :  Disabled,  timer 300
Nonstop Routing:             Disabled
Graceful Restart:            Disabled,  timer 120
Graceful Restart Helper:     Enabled
LDP-SYNC: Globally enabled, Hold-down time 100 sec
```

In this example, the LDP-SYNC fields displays the enabling of LDP-IGP synchronization and the hold-down time.

12. Verify the LDP-IGP synchronization configuration with IS-IS.

```
device# show running config router isis
 net 11.0010.0100.1002.00
 log adjacency
 address-family ipv4 unicast
 ldp-sync
  metric-style wide
  ldp-sync
  ldp-sync hold-down 100
 redistribute connected level-1
 redistribute static
 exit-address-family

 address-family ipv6 unicast
 ldp-sync
  multi-topology transition
 exit-address-family

end
```

The following example shows the previous steps to configure to globally enable MPLS LDP-IGP synchronization with OSPF and IS-IS.

```
device(config)# router ospf
device(config-router-ospf-vrf-default-vrf)# ldp-sync
device(config-router-ospf-vrf-default-vrf)# ldp-sync hold-down 100
device(config-router-ospf-vrf-default-vrf)# exit
device(config)# router isis
device(config-router-isis)# address-family ipv4 unicast
device(config-router-isis-ipv4u)# metric-style wide
device(config-router-isis-ipv4u)# ldp-sync
device(config-router-isis-ipv4u)# ldp-sync hold-down 100
```

Enabling MPLS LDP-IGP synchronization on an interface

You can enable LDP-IGP synchronization on an interface that belongs to an OSPF or IS-IS process and override global LDP-IGP synchronization.

Perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enter the configuration mode to an Ethernet port.

```
device(config)# interface ethernet 1/1
```

3. Enable IS-IS on an interface.

```
device(config-if-eth-1/1)# ip router isis
```

4. Enable LDP-IGP synchronization with IS-IS on the interface.

```
device(config-if-eth-1/1)# isis ldp-sync enable
```

5. Configure the OSPF area on the interface.

```
device(config-if-eth-1/1)# ip ospf area 0.0.0.0
```

6. Enable LDP-IGP synchronization with OSPF on the interface.

```
device(config-if-eth-1/1)# ip ospf ldp-sync enable
```

7. Verify LDP-IGP synchronization with IS-IS on the interface.

```
device# show isis interface
Total number of IS-IS Interfaces: 5

Interface: eth 1/1
Circuit State: UP Circuit Mode: LEVEL-1-2
Circuit Type: BCAST Passive State: FALSE Circuit Number: 3, MTU: 1500
Level-1 Auth-mode: None
Level-2 Auth-mode: None
Level-1 Metric: 10, Level-1 Priority: 64
Level-1 Hello Interval: 10 Level-1 Hello Multiplier: 3
Level-1 Designated IS: R2-03 Level-1 DIS Changes:
Level-2 Metric: 10, Level-2 Priority: 64
Level-2 Hello Interval: 10 Level-2 Hello Multiplier: 3
Level-2 Designated IS: R2-03 Level-2 DIS Changes: 2
Next IS-IS LAN Level-1 Hello in 7 seconds
Next IS-IS LAN Level-2 Hello in 3 seconds
Number of active Level-1 adjacencies: 0
Number of active Level-2 adjacencies: 0
Circuit State Changes: 1 Circuit Adjacencies State Changes: 0
Rejected Adjacencies: 0
Circuit Authentication L1 failures: 0
Circuit Authentication L2 failures: 0
Bad LSPs: 0
Control Messages Sent: 40 Control Messages Received: 0
Hello Padding: Enabled
IP Enabled: TRUE IP Addresses:
   10.30.30.2/28
IPv6 Enabled: FALSE
MPLS TE Enabled: FALSE
LDP-SYNC: Enabled, State:
```

8. Verify LDP-IGP synchronization with OSPF on the interface.

```
device# show ip ospf
OSPF Version Version 2
Router Id 10.1.1.2
ASBR Status No
ABR Status No (0)
Redistribute Ext Routes from
Initial SPF schedule delay 0 (msecs)
Minimum hold time for SPFs 0 (msecs)
Maximum hold time for SPFs 0 (msecs)
External LSA Counter 0
External LSA Checksum Sum 00000000
Originate New LSA Counter 9
Rx New LSA Counter 6
External LSA Limit 174762
Database Overflow Interval 0
Database Overflow State : NOT OVERFLOWED
RFC 1583 Compatibility : Enabled
Slow neighbor Flap-Action : Disabled, timer 300
Nonstop Routing: Disabled
Graceful Restart: Disabled, timer 120
Graceful Restart Helper: Enabled
LDP-SYNC: Globally enabled, Hold-down time 66 sec
Interfaces with LDP-SYNC enabled:
eth 1/1
```

The following example shows the previous steps.

```
device# configure terminal
device(config)# interface e 1/1
device(config-if-eth-1/1)# ip router isis
device(config-if-eth-1/1)# isis ldp-sync enable
device(config-if-eth-1/1)# ip ospf area 0.0.0.0
device(config-if-eth-1/1)# ip ospf ldp-sync enable
```

Configuring the receive label silence and EOL timers for LDP-IGP synchronization

You can configure the receive label silence and EOL timers for LDP-IGP synchronization.

Perform the following steps to configure the timers.

1. In privileged EXEC mode, enter global configuration mode.

   ```
   device# configure terminal
   ```

2. Enable MPLS on the device.

   ```
   device(config)# router mpls
   ```

3. Access LDP configuration mode.

   ```
   device(config-router-mpls)# ldp
   ```

4. Define the length of the receive label silence timer.

   ```
   device(config-router-mpls-ldp)# rx-label-silence-time 60000
   ```

   In this example, the length of time for the receive label silence timer is set to 60000 milliseconds. Possible values are from 100 to 60000. The default value is 1000. When labels are not received from the peer for a short period of time, the session is declared In Sync. When a label is received from a peer, then the receive label silence timer is reset.

5. Enable the end-of-lib configuration mode.

   ```
   device(config-router-mpls-ldp)# eol
   ```

   The end-of-lib mode contains all the attributes of the end of lib capability and notification. Also, enabling the end-of-lib mode determines whether the two RFCs 5561 and 5919 are enabled by the LSR.

6. Set the length of the EOL notification timer.

   ```
   device(config-router-mpls-ldp-eol)# notification-timer 80000
   ```

   In this example, the length of time for the EOL notification timer is set to of 80000 milliseconds. Possible values are from 100 to 120000. The default value is 60000.

7. Set the length of the EOL transmit label silence timer.

   ```
   device(config-router-mpls-ldp-eol)# tx-label-silence-timer 2000
   ```

   In this example, the length of time for the EOL transmit label silence timer is set to of 2000 milliseconds. Possible values are from 100 to 60000. The default value is 1000.

8. Verify the configuration.

```
device# show mpls ldp
Label Distribution Protocol version 1
 LSR ID: 10.1.7.1, using Loopback 1 (deleting it stops LDP)
 Hello interval: Link 5 sec, Targeted 15 sec
 Hold time value sent in Hellos: Link 15 sec, Targeted 45 sec
 Keepalive interval: 6 sec, Hold time multiple: 6 intervals
 Load sharing: 8
 Tunnel metric: 0
 FEC used for auto discovered peers: current 129, configured 129
 End of lib: Enabled, Notification time: 80000 ms, tx label silence time: 2000ms
 Rx label silence time: 80000ms
 Graceful restart: enabled
  Reconnect time: 120 seconds, Max peer reconnect time: 120 seconds
  Recovery time: 120 seconds, Max peer recovery time: 120 seconds
  Forwarding state holding timer: not running
 Label Withdrawal delay: 60 seconds (Default)
```

The **show mpls ldp** command displays the configuration of the EOL parameters.

The following example shows the previous steps of the configuration.

```
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# rx-label-silence-time 30000
device(config-router-mpls-ldp)# eol
device(config-router-mpls-ldp-eol)# notification-timer 80000
device(config-router-mpls-ldp-eol)# tx-label-silence-timer 2000
```

Displaying LDP-IGP synchronization information

In addition to configuration information, you can display the following LDP-IGP synchronization information.

- LDP-IGP synchronization for an LDP session
- Whether LDP-IGP synchronization is enabled on the interface
- The current cached LDP-IGP synchronization state

Displaying LDP-IGP synchronization on a session

To display LDP-IGP synchronization, use the **show mpls ldp session** command. When the timers are running, the remaining time appears without a refresh as well as the static value of the timer.

```
device# show mpls ldp session 10.7.7.2 0
Peer LDP ID: 10.7.7.2:0, Local LDP ID: 10.1.7.1:0, State: Operational
  Adj: Link, Role: Passive, Next keepalive: 1 sec, Hold time left: 31 sec
  Keepalive interval: 6 sec, Max hold time: 36 sec
  Up time: 16 min 46 sec
  TCP connection: 10.1.7.1:646--10.7.7.2:9004, State: ESTABLISHED
  Neighboring interfaces: Eth 2/2
  Next-hop addresses received from the peer:4
    10.1.1.14    10.1.2.14    10.3.4.5    10.7.7.2
  IGP Sync:
   Unrecognized Notification Capability: Local: On, Remote: On
   Local State: In-sync, RemoteState: In-sync
   Rx label silence time: 1000 ms, Timer not running
  Graceful restart: enabled
    Peer reconnect time(msec): 120000, peer recovery time(msec): 0
    State: not started
```

Displaying whether LDP-IGP synchronization is enabled on the interface

To display whether LDP-IGP synchronization is enabled on the interface, use the **show mpls ldp interface** command.

```
device# show mpls ldp int e2/16
e2/16, label-space ID: 0
  Nbr count: 1
  Hello interval: 5 sec, next hello: 2 sec
  Hello timeout: 15 sec
  Current IGP Sync state: In Sync
```

Displaying the current cached LDP-IGP synchronization state

To display the current cached LDP-IGP synchronization state, use the **show mpls interface** command.

```
device# show mpls int e2/2
  Admin: Up    Oper: Up
  Maximum BW: 0 kbps, maximum reservable BW: 0 kbps
  Admin group: 0x00000000
  Reservable BW [priority] kbps:
    [0] 0    [1] 0    [2] 0    [3] 0 [4] 0    [5] 0    [6] 0    [7] 0
  Last sent reservable BW [priority] kbps: [0] 0    [1] 0    [2] 0    [3] 0
    [4] 0    [5] 0    [6] 0    [7] 0
  LDP tunnel count: 0
  Current IGP Sync State: not-in-sync
```

LDP Graceful Restart

LDP Graceful Restart (GR) helps minimize MPLS traffic loss when an LDP component is restarting in a router that is capable of preserving its MPLS forwarding states across restart. LDP GR is based on RFC 3478 (Graceful Restart mechanism for Label Distribution Protocol).

LDP GR works between a router and its neighbor and its capability must be advertised when sending an LDP Initialization message. An LDP restart triggered by an MP failover due to a fault of the active MP or a command-initiated switchover is the only scenario where the MPLS forwarding state is preserved.

The router can also support LDP GR in helper-only mode. In this mode, a router does not preserve its forwarding entries on a LDP GR restart. It indicates to its peers that forwarding state is not preserved by sending an initialization message with the Reconnect Time and the Recovery Time set to zero (0) in FT session TLV. However, it can help a neighboring router recover its forwarding entries when the neighbor is going through restart.

A LDP GR enabled router goes into helper-only mode (GR helper) when any of the following events occur on the router's neighbors.

- MP failover occurs
- HLOS upgrade occurs
- Remove and re-add of the MPLS configuration
- TCP communication broken (such as, session keepalive timer expires)
- UDP communication broken (for example, an adjacency goes down)
- Restarting LDP component by disabling and enabling the loopback
- Restarting a LDP session by issuing the **clear mpls ldp neighbor** command

In helper-only mode, the LDP GR procedure works at the session level. Any of the previous events causes the helper to detect session down and start the GR procedure. The operation of the GR helper is the same independent of what has happened on the restarting LSR that triggers the GR procedure.

When LDP GR is enabled on a router, the configuration does not apply to the current sessions. The LDP GR configuration is applied for the new sessions brought up after the configuration is added.

NOTE
LDP GR supports hitless IP over MPLS.

LSR restarting procedure

After an LSR restarts its LDP components, when its MPLS forwarding state is not preserved as in the case of the routers in helper-only mode, it sends out the FT TLV with Recovery Time set to 0 in the LDP Initialization message to its neighbor.

When the MPLS forwarding state has been preserved across the restart, the LSR does the following:

1. Start the Forwarding State Holding timer.
2. Mark all the MPLS forwarding entries as stale.
3. Set the Recovery Time to the current value of the Forwarding State Holding timer when it sends out LDP Initialization message to its neighbor.
   - When the timer has not expired, the LSR uses the labels and next-hop information received from the neighbor to look up and clear the stale flag for the corresponding label-FEC entries.
   - When the timer has expired, all the entries that are still marked as stale are deleted and the LDP GR procedure is completed.

GR helper LSR restarting procedure

When the LSR detects that its LDP session with a neighbor is down and the neighbor is capable of preserving its forwarding state, the LSR does the following:

1. Retains the label-FEC bindings received by way of the session and marks them as stale.
2. Starts the Reconnect timer with the timeout value set to the lesser of the peer FT Reconnect Timeout and the locally configured maximum Reconnect timeout.
3. Attempts to re-establish the LDP session with the neighbor using the normal LDP procedure. All the stale label-FEC bindings are deleted when either condition is true:
   - The Reconnect timer has expired and the LDP session to the neighbor is not established.
   - The LSR receives FT TLV in the Initialization message from the neighbor and the FT Recovery Time is set to 0.

   After the session is re-established, the LDP GR helper resends the Label Mappings to its neighbor. For the stale label-FEC bindings received from the neighbor, they are recovered during the recovery period which is set to the lesser of the peer Recovery Timeout and the locally configured maximum recovery time. If the stale entries are not recovered after the Recovery Timer has expired, they are deleted.

Graceful restart scenarios

Graceful restart includes the following scenarios:

- Re-advertise label to its upstream neighbors—When the restarting router, acting as a transit LSR, can recover a FEC based on the Label Mapping it receives from its GR helper, and the local forwarding state successfully, it re-advertises the same label to all of its upstream neighbors.

As part of supporting GR, the Label Management component also makes sure that those labels that are used to advertise to upstream neighbors before GR happens is not re-used for the new LSP coming up while GR is in-progress. However, when the previously used label is released because the LSP has gone down during GR, the label can be re-allocated for the new LSP.

- Clearing MPLS LDP neighbors— For the **clear mpls ldp neighbor** command, the configured reconnect and recovery timer values is sent to the peer when both are configured with LDP graceful restart. Note that in this scenario, both routers are acting in helper-only mode. Therefore, after the session comes back up, both routers exchange their bindings and go through the recovery procedure. There is no traffic loss when the reconnect and the recovery timers do not time out.

  On a router configured for helper-only mode, the **clearing mpls ldp neighbor** command results in immediate reconnect timeout at the remote end. Therefore, in this scenario all bindings at the remote end associated with the session are deleted due to reconnect time out. On the local node the recovery timer of zero results in immediate clearing of the forwarding entries.

Ingress LSR specific processing

VPLS supports failover and must preserve its forwarding state when the LDP tunnel is used to carry VPLS traffic until the GR has finished. LDP GR attempts to recover both the tunnel labels and the VC labels. In the case where a VC label cannot be recovered, the corresponding Pseudo Wire (PW) is brought down after GR has finished. In the case where the LDP tunnel cannot be recovered, all the PWs using the LDP tunnel is brought down due to tunnel down event.

LDP tunnels are preserved across failover and application can use them to provide hitless support.

Transit LSR specific processing

For those LDP cross-connects that can be recovered as part of LDP GR, there is no traffic loss for those application using those tunnels if and only if the GR helper (for example, a downstream neighbor) re-advertises the same label and upstream neighbor also support LDP GR procedure as well.

LDP GR preserves the LDP transit cross-connects. It minimizes the traffic loss of any application that uses an LDP tunnel as its transport mechanism from the transit LSR perspective.

LDP ECMP (transit only)

The ability to preserve the LDP ECMP transit cross-connects depends on the route information received from the RIB Manager during the recovery phase. In the case where the number of ECMP provided by the RIB Manager for a route is larger than the LDP load-sharing configuration (for example, the IP load-sharing configuration is larger than LDP load-sharing configuration), the paths are preserved as long as the route provided by the RIB Manager, before recovery time expires, contains the installed paths.

Configuring LDP graceful restart (GR)

By default LDP GR is disabled. You can globally enable it. When LDP GR enabled, the router waits until it receives an LDP Initialization message from its neighbor to know whether it must delete its states or start the LDP GR recovery procedure. Also, LDP GR is applicable to all LDP sessions regardless of the adjacency type exists between the neighbors.

NOTE
The following commands only take effect on newly created sessions. For existing sessions, it is required that the sessions be restarted for the new configuration to take effect.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

4. Enable LDP GR and access GR configuration mode.

```
device(config-router-mpls-ldp)# graceful-restart
```

5. If you are configuring the router as an LSR acting as a GR helper, use the **helper-only** command.

```
device(config-router-mpls-ldp-gr)# helper-only
```

   In helper mode, the commands for reconnect time and recovery time are rejected with informational messages. If you do not configure this command, the router acts as a restarting LSR.

6. Specify the session reconnect time.

```
device(config-router-mpls-ldp-gr)# graceful-restart reconnect-time 150
```

7. Specify the amount of time that this device retains its MPLS forwarding state across LDP GR.

```
device(config-router-mpls-ldp-gr)# graceful-restart recovery-time 240
```

8. Specify the maximum amount of time to wait for a neighbor to reconnect.

```
device(config-router-mpls-ldp-gr)# graceful-restart max-neighbor-reconnect-time 150
```

9. Specify the maximum amount of time to wait for a neighbor to recover.

```
device(config-router-mpls-ldp-gr)# graceful-restart max-neighbor-recovery-time 240
```

10. Verify that graceful restart is enabled.

```
device# show mpls ldp
Label Distribution Protocol version 1
  LSR ID: 10.210.210.21, using Loopback 1 (deleting it stops LDP)
  Hello interval: Link 5 sec, Targeted 15 sec
  Hold time value sent in Hellos: Link 15 sec, Targeted 45 sec
  Keepalive interval: 6 sec, Hold time multiple: 6 intervals
  Load sharing: 8
  Tunnel metric: 0
  FEC used for auto discovered peers: current 129, configured 129
  Graceful restart: enabled
    Reconnect time: 150 seconds, Max peer reconnnect time: 150 seconds
    Recovery time: 240 seconds, Max peer recovery time: 240 seconds
    Forwarding state holding timer: not running
```

The following example shows the previous steps to configure LDP GR.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# graceful-restart
device(config-router-mpls-ldp-gr)# graceful-restart reconnect-time 150
device(config-router-mpls-ldp-gr)# graceful-restart recovery-time 240
device(config-router-mpls-ldp-gr)# graceful-restart max-neighbor-reconnect-time 150
device(config-router-mpls-ldp-gr)# graceful-restart max-neighbor-recovery-time 240
device(config-router-mpls-ldp-gr)# helper-only
```

The "helper-only" configuration is mutually exclusive with the recovery and reconnect time configuration. If any of these two timers are configured to non-default value, the helper-only configuration cannot be applied.

```
device(config-router-mpls-ldp-gr)# helper-only
%Error: Unconfigure LDP GR timers before configuring GR Helper mode
device(config-router-mpls-ldp-gr)# no recovery-time
Possible completions:
  <cr>
device(config-router-mpls-ldp-gr)# no recovery-time
device(config-router-mpls-ldp-gr)# helper-only
device(config-router-mpls-ldp-gr)#

device(config-router-mpls-ldp-gr)# helper-only
device(config-router-mpls-ldp-gr)#
device(config-router-mpls-ldp-gr)#
device(config-router-mpls-ldp-gr)# recovery-time 150
%Error: Unconfigure LDP GR Helper mode before configuring GR timers
device(config-router-mpls-ldp-gr)# no helper-only
device(config-router-mpls-ldp-gr)# recovery-time 150
device(config-router-mpls-ldp-gr)#
```

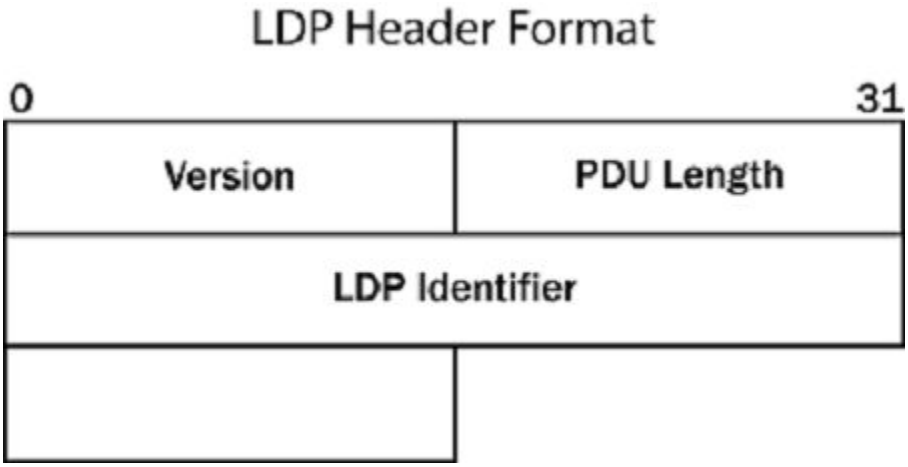
Configurable LDP router ID

LDP uses LDP messages to communicate between LDP peers for the correct functioning of LDP. All LDP messages contains a LDP header which is composed of LDP version, length of message, LDP ID, and is followed by a message. The LDP ID for LDP is composed of the LSR-ID and label space. The LSR ID is the first available loopback interface address. However, you can specify an IP address of your choice to use as the LSR ID for the LDP identifier.

By default, Extreme routers select the first valid and operationally UP IP address among all the enabled loopback interfaces as LSR ID for LDP. When the IP address or loopback interface that is used as the LSR ID goes down, LDP selects the next operationally UP IP address among all enabled loopback interface as the LSR ID. Otherwise, LDP will be down.

When no valid IP address is available to be selected as the LSR ID, LDP continues to remain disabled until a valid IP address is configured on an enabled loopback interface.

FIGURE 29 LDP Header format



LDP uses a configured IP address as the LSR ID only when this IP address is configured on one of the enabled loopback interfaces. After you configure an LSR ID with a valid IP address, LDP must use the configured value as the LSR ID and restarts to use the new address.

When this IP address is not configured in the enabled state on any of the loopback interfaces, LDP continues in the disabled state. LDP is enabled as soon as this IP address is configured on one of the enabled loopback interfaces.

When you disable the feature, the LSR ID selection procedure falls back to default behavior of selecting an LSR ID for LDP when LDP is enabled.

Limitations

- You cannot configure value 0.0.0.0. If you try to configure the feature with this value, the feature rejects the configuration.
- You can only configure IPv4 addresses for the LSR ID.

Configuring the LDP router ID

By default, the LSR-ID is the first available loopback interface address. However, you can specify an IP address of your choice to use as the LSR-ID for the LDP identifier.

Ensure that you configure the LSR ID IP address is an operationally UP IP address on an enabled loopback interface.

Perform the following steps to configure an LSR ID.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable MPLS on the device.

```
device(config)# router mpls
```

3. Access LDP configuration mode.

```
device(config-router-mpls)# ldp
```

4. Configure an IP address for the LSR ID.

```
device(config-router-mpls-ldp)# lsr-id 10.22.22.22
```

   The configured IP address selected as the LSR ID for LDP is an operationally UP IP address on an enabled loopback interface. You can configure only an IPv4 address.

The following example is the configuration of the previous steps.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# ldp
device(config-router-mpls-ldp)# lsr-id 10.22.22.22
```

Displaying LDP information and statistics

The **show mpls ldp** command and options allows you to display LDP information and statistics.

Displaying the LDP version

To display the LDP version number, the LSR ID and loopback number, and the LDP hello interval and hold time, enter the **show mpls ldp** command.

```
device# show mpls ldp
Label Distribution Protocol version 1
  LSR ID: 1.2.3.4 , using Loopback 1 (deleting it will stop LDP)
  Hello interval: Link 5 sec, Targeted 15 sec
  Hold time value sent in Hellos: Link 15 sec, Targeted 45 sec
  Keepalive interval: 6 sec, Hold time multiple: 6 intervals
  Keepalive timeout: 36 sec
  Load sharing: 8
  Advertise FECs for prefix-list: "ldp-route-injection"
  Tunnel metric: 0
  FEC used for auto discovered peers: current 129, configured 129
  End of LIB: Disabled, Notification time: 60000 ms, tx label silence time: 1000 ms
  Rx label silence time: 1000 ms
  Graceful restart: enabled as helper-only
    Reconnect time: 0 seconds, Max peer reconnect time: 120 seconds
    Recovery time: 0 seconds, Max peer recovery time: 120 seconds
    Forwarding state holding timer: not running
  Label Withdrawal delay: 60 seconds (Default)
```

Displaying LDP-created LSPs information

To display information about active LDP-created LSPs for which this device is an ingress, transit, or egress LSR, use the **show mpls ldp path** command. The command shows information about an LSP created through LDP including the incoming and outgoing labels applied to packets in each LSP.

```
device# show mpls ldp path
Destination route   Upstr-session(label)       Downstr-session(label, intf)
10.2.2.2/32         10.3.3.3:0(1024)           10.2.2.2:0(3,Eth 2/10)
                    10.2.2.2:0(1024)           10.2.2.2:0(3,Eth 2/10)
10.3.3.3/32         10.3.3.3:0(1026)           10.3.3.3:0(3,Eth 2/20)
                    10.2.2.2:0(1026)           10.3.3.3:0(3,Eth 2/20)
```

Displaying LDP tunnel LSP information

To display information about LDP-created LSPs for which this device is the ingress LER including the total number of tunnels, use the **show mpls ldp tunnel** command.

```
device# show mpls ldp tunnel
Total number of LDP tunnels : 3
                  Oper     Tunnel     Outbound
To                State    Intf       Intf
10.22.22.22/32    UP       tnl2       Eth 2/3
10.33.33.33/32    UP       tnl3       Eth 2/3
10.44.44.44/32    UP       tnl4       Ve 55
```

If you include the tunnel destination IP prefix with the command, it displays a single tunnel entry for a specified prefix.

```
device# show mpls ldp tunnel 10.22.22.22
LDP tunnel tnl2, to 10.22.22.22/32
  Tunnel index: 2, metric: 0, status: UP
  Outgoing interface: Eth 2/3, Next-hop ip: 10.55.55.55/32
```

If you include the **detail** option with the command, it displays a single tunnel entry for a specified prefix.

```
show mpls ldp tunnel detail
Total number of LDP tunnels : 3
LDP tunnel tnl2, to 10.22.22.22/32
  Tunnel index: 2, metric: 0, status: UP
  Outgoing interface: Eth 2/3, Next-hop ip: 10.55.55.55/32
LDP tunnel tnl3, to 10.33.33.33/32
  Tunnel index: 3, metric: 0, status: UP
  Outgoing interface: Eth 2/3, Next-hop ip: 10.55.55.55/32
LDP tunnel tnl4, to 10.44.44.44/32
  Tunnel index: 4, metric: 0, status: UP
  Outgoing interface: Ve 55, Next-hop ip: 10.55.55.55/32
```

Displaying the contents of the LDP database

To display the contents of the LSRs LDP Label Information Base, use the **show mpls ldp database** command. This database contains all labels that it learned from each of its LSR peers and that it sent to its LDP peers.

```
# show mpls ldp database
Session 10.210.210.21:0 - 10.2.2.2:0
 Downstream label database:
   Label     Prefix             State
 Upstream label database:
  Label      Prefix
  1024       10.125.125.25/32(Stale)
  3          10.210.210.21/32(Stale)
  1025       10.220.220.22/32(Stale)
Session 10.210.210.21:0 - 10.220.220.22:0
 Downstream label database:
   Label     Prefix              State
   3         10.220.220.22/32    Installed
   1024      10.125.125.25/32    Installed
   983097    VC-FEC              Retained
 Upstream label database:
   Label     Prefix
   3         10.210.210.21/32
   983040    VC-FEC
```

Displaying LDP session information

To display information about the LDP session between this LSR and its LDP peers, enter the **show mpls ldp session** command. The following examples show Graceful Restart related information received from the neighbor.

When Graceful Restart starts, it also shows the current state.

```
device# show mpls ldp session
 Number of link LDP sessions: 2
 Number of Operational link LDP sessions: 2
 Number of targeted LDP sessions: 1
 Number of Operational targeted LDP sessions: 1

Peer LDP ID       State           Adj Used     My Role     Max    Hold    Time Left
10.22.22.22:0     Operational     Link         Passive     36             33
10.33.33.33:0     Operational     Targeted     Passive     36             34
10.44.44.44:0     Operational     Link         Passive     36             34
```

The following example shows the reconnect timer running.

```
device# show mpls ldp session 10.2.2.2:0
Peer LDP ID: 10.2.2.2:0, Local LDP ID: 10.210.210.21:0, State: Restarting
   Graceful restart: enabled
     Peer reconnect time(msec): 120000, peer recovery time(msec): 120000
     Reconnect time in use(msec): 120000, remaining time(msec): 32300
     State: reconnecting
```

The following example shows the recovery timer running.

```
device# show mpls ldp session 10.11.11.1
Peer LDP ID: 10.11.11.1:0, Local LDP ID: 10.22.22.2:0, State: Operational
  Adj: Link, Role: Active, Next keepalive: 0 sec, Hold time left: 30 sec
  Keepalive interval: 6 sec, Max hold time: 36 sec
  Up time: 56 sec
  TCP connection: 10.22.22.2:9001--10.11.11.1:646, State: ESTABLISHED
  Neighboring interfaces: (targeted), Eth 1/1
  Next-hop addresses received from the peer:4
  10.9.1.1    10.10.1.1    10.11.11.1    10.11.11.12
  Graceful restart: enabled
  Peer reconnect time(msec): 120000, peer recovery time(msec): 120000
  Recovery time in use(msec): 120000, remaining time(msec): 103000
  State: recovering
```

The following example displays information about the configured and peer proposed keepalive timeout, and FEC label information including the count of installed, received, and filtered FEC labels.

```
device# show mpls ldp session 10.0.0.13
Peer LDP ID: 10.0.0.13:0, Local LDP ID: 10.0.0.1:0, State: Operational
  Adj: Link, Role: Passive, Next keepalive: 3 sec, Hold time left: 33 sec
  Keepalive interval: 6 sec, Max hold time: 36 sec
  Configured keepalive timeout: 36 sec
  Peer proposed keepalive timeout: 36 sec
  Up time: 4 min 56 sec
  TCP connection: 10.0.0.1:646--10.0.0.13:9098, State: ESTABLISHED
  Neighboring interfaces: Eth 2/1
  Next-hop addresses received from the peer:11
  10.0.0.13 192.168.3.2 192.168.6.2 192.168.9.2 192.168.12.2
  192.168.16.2 192.168.19.1 192.168.20.1 192.168.28.2 192.168.40.2
  192.168.45.2
  IGP Sync:
    Unrecognized Notification Capability: Local: Off, Remote: Off
    Local State: In-sync, RemoteState: -
    Rx label silence time: 1000 ms, Timer not running
  Graceful restart: disabled
  Number of FECs Received from peer: 4013
  Number of FECs installed from peer: 4005
  Number of FECs filtered from peer(in/out): 0/0
```

Displaying LDP neighbor connection information

To display information about the connection between this LSR and its LDP-enabled neighbors, use the **show mpls ldp neighbor** command.

```
device# show mpls ldp neighbor
Number of link neighbors: 2
Number of targeted neighbors: 1

Nbr Transport   Interface   Nbr LDP ID   Max Hold    Time Left
10.1.1.1        Eth 4/1     10.1.1.1:0   15          14
10.5.5.5        Eth 3/2     10.5.5.5:0   15          11
10.4.4.4        (targeted)  10.4.4.4:0   15          13
```

To display the adjacency uptime since the LDP adjacency is established, use the **show mpls ldp neighbor detail** command.

```
device# show mpls ldp neighbor detail
Nbr Transport Addr: 10.22.22.1, Interface: e1/1, Nbr LDP ID: 10.22.22.1:0
 MaxHold: 15 sec, Time Left: 13 sec, Up Time: 14 hr 13 min 6 sec
 Configured Hold time: 15 sec, Neighbor Proposed Hold Time: 15 sec

Nbr Transport Addr: 10.22.22.1, Interface: e1/2, Nbr LDP ID: 10.22.22.1:0
 MaxHold: 15 sec, Time Left: 12 sec, Up Time: 14 hr 13 min 1 sec
 Configured Hold time: 15 sec, Neighbor Proposed Hold Time: 15 sec

Nbr Transport Addr: 10.33.33.1, Interface: e1/3, Nbr LDP ID: 10.33.33.1:0
 MaxHold: 15 sec, Time Left: 13 sec, Up Time: 14 hr 13 min 5 sec
 Configured Hold time: 15 sec, Neighbor Proposed Hold Time: 15 sec

Nbr Transport Addr: 10.33.33.1, Interface: targeted, Nbr LDP ID: 10.33.33.1:0
 MaxHold: 15 sec, Time Left: 10 sec, Up Time: 14 hr 13 min 3 sec
 Configured Hold time: 15 sec, Neighbor Proposed Hold Time: 15 sec
```

Displaying the LDP packet statistics

To display a packet statistics for packet types and packet errors, use the **show mpls ldp statistics** command.

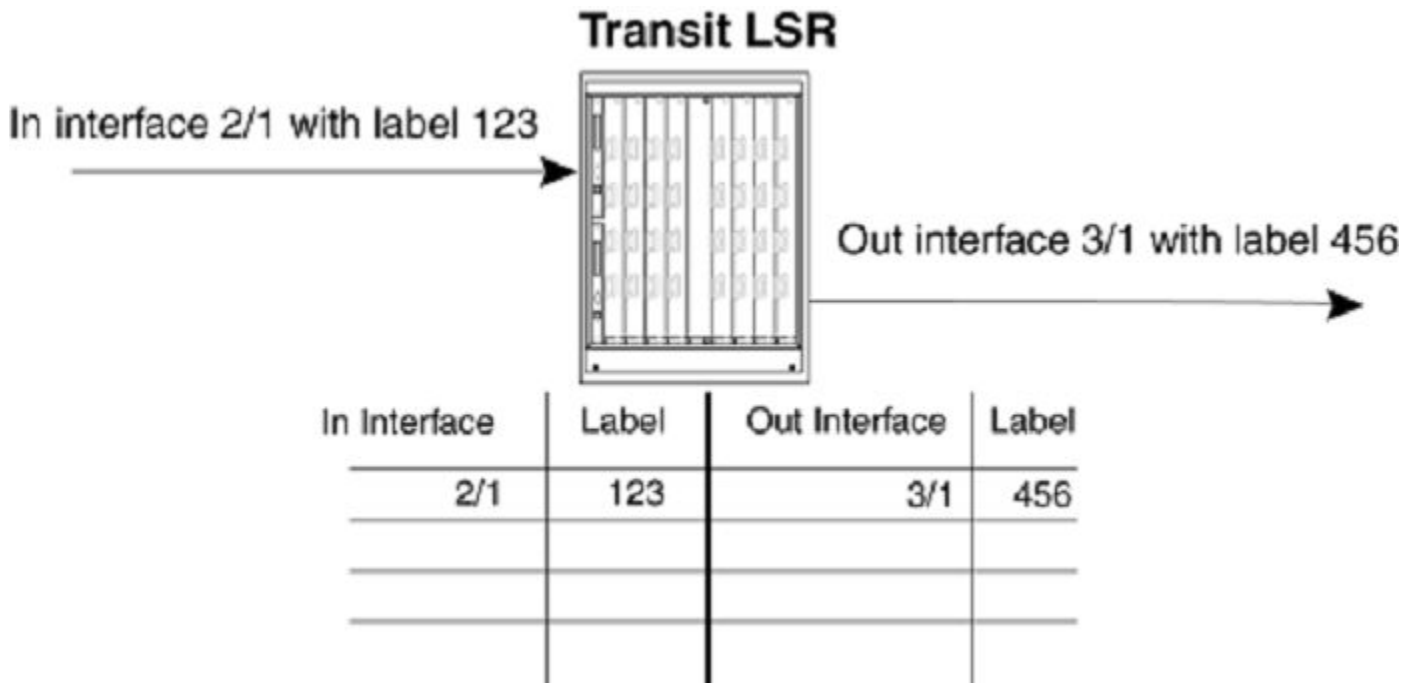
```
device# show mpls ldp statistics
                  Total                     Since last clear
MessageType       Sent          Received      Sent          Received
  Notify          3             0             3             0
  Link Hello      112254        112253        112254        112253
  Targeted Hello  6824          6775          6824          6775
  Initialize      7             8             7             8
  Keepalive       42470         43875         42470         43875
  Addr            42            11            42            11
  AddrWdrw        25            13            25            13
  LabelMap        26851         1654          26851         1654
  LabelReq        0             0             0             0
  LabelWdrw       9228          18            9228          18
  LabelRel        18            32            18            32
  LabelAbReq      0             0             0             0
  Unknown         0             0             0             0

Errors                        Total                Since last clear
  Rcv pkt bad pdu length      0                    0
  Rcv pkt bad msg length      0                    0
  Rcv pkt bad tlv length      0                    0
  Rcv pkt notify unkn tlv     0                    0
  Rcv pkt notify unkn addrfam 0                    0
  Rcv pkt missing tlv         0                    0
  Rcv pkt incorrect tlv       0                    0
  Rcv pkt malformed tlv       0                    0
  Rcv pkt bad traffic parm    0                    0
  Rcv pkt partial pdu         0                    0
  Rcv pkt internal error      0                    0
  TCP get send pkt error      0                    0
```

```
TCP send error              0                   0
TCP memory fail             0                   0
```

Configuration example of LDP-enabled LSRs

The following figure illustrates a configuration example of three LDP-enabled LSRs.

FIGURE 30 LDP-enabled LSR configuration



The following example provides the configuration of the LSR on each router.

Router R1

```
R1(config)# interface loopback 1
R1(config-lbif-1)# ip address 10.1.1.1/32
R1(config-lbif-1)# exit
R1(config)# router mpls
R1(config-router-mpls)# mpls-interface ethernet 2/10
R1(config-router-mpls-if-ethernet-2/10)# ldp-params
R1(config-router-mpls-if-ethernet-2/10-ldp-params)# ldp-enable
R1(config-router-mpls)# exit
R1(config)# ip route 10.2.2.2/32 10.1.1.2
R1(config)# ip route 10.3.3.3/32 10.1.1.2
R1(config)# route-only
R1(config)# interface ethernet 2/10
R1(config-if-ethernet-2/10)# ldp-params
R1(config-if-ethernet-2/10-ldp-params)# enable
R1(config-if-ethernet-2/10-ldp-params)# ip address 10.1.1.1/24
R1(config-if-ethernet-2/10-ldp-params)# exit
R1(config)# interface ethernet 2/20
R1(config-if-ethernet-2/20)# ldp-params
R1(config-if-ethernet-2/20-ldp-params)# enable
R1(config-if-ethernet-2/20-ldp-params)# ip address 10.1.1.1/24
```

Router R2

```
R2(config)# interface loopback 1
R2(config-lbif-1)# ip address 10.2.2.2/32
R2(config-lbif-1)# exit
R2(config)# router mpls
R2(config-router-mpls)# mpls-interface ethernet 2/10
R2(config-router-mpls-if-ethernet-2/10)# ldp-params
R2(config-router-mpls-if-ethernet-2/10-ldp-params)# ldp-enable
R2(config-router-mpls-if-ethernet-2/10-ldp-params)# exit
R2(config)# ip route 10.1.1.1/32 10.1.1.1
R2(config)# ip route 10.3.3.3/32 10.1.1.1
R2(config)# route-only
R2(config)# interface ethernet 2/20
R2(config-if-ethernet-2/20)# ldp-params
R2(config-if-ethernet-2/10-ldp-params)# enable
R2(config-if-2/20)# ip address 10.1.1.2/24
R2(config-if-2/20)# exit
```

Router R3

```
R3(config)# interface loopback 1
R3(config-lbif-1)# ip address 10.3.3.3/32
R3(config-lbif-1)# exit
R3(config)# router mpls
R3(config-router-mpls)# mpls-interface ethernet 2/10
R3(config-router-mpls-if-ethernet-2/10)# ldp-params
R3(config-router-mpls-if-ethernet-2/10-ldp-params)# ldp-enable
R3(config-router-mpls-if-ethernet-2/10-ldp-params)# exit
R3(config)# ip route 10.1.1.1/32 10.1.1.1
R3(config)# ip route 10.2.2.2/32 10.1.1.1
R3(config)# route-only
R3(config)# interface ethernet 2/20
R3(config-if-ethernet-2/20)# ldp-params
R3(config-if-ethernet-2/20-ldp-params)# enable
R3(config-if-ethernet-2/20-ldp-params)# 10.1.1.2/24
R3(config-if-ethernet-2/20-ldp-params)# exit
```

IP over MPLS

BGP shortcuts using next-hop MPLS

By default, in a typical configuration, BGP considers only IP routes for next-hop resolution when building a routing table. When an MPLS network uses BGP to propagate routes, BGP may consider whether the MPLS tunnels are viable routes.

You can globally enable BGP shortcuts using next-hop MPLS on an Extreme device to configure BGP to use an MPLS tunnel as the preferred next-hop route to a destination network when the tunnel is available. You can also configure BGP to include LSP metrics for best-route computations. You can also configure BGP to use the IGP metric of LSP tunnels for best-route computations. When the BGP attempt at route next-hop resolution through MPLS tunnels is unsuccessful, the Extreme device uses the IPv4 routing table to resolve the route.

When you configure BGP shortcuts using next-hop MPLS on an MPLS edge router, BGP computes routes to destinations available through other edge routers. When BGP determines that a route is available through an edge router that is reachable through an MPLS tunnel, a BGP shortcut directs BGP to place the MPLS tunnel in the routing table as the preferred BGP route.

NOTE

When an MPLS LSP tunnel to a BGP next hop is available, BGP always prefers the LSP tunnel to resolve a BGP next hop if the **next-hop-mpls** command is enabled. Mixed ECMP between MPLS LSP and native IGP path are not supported for a given BGP next hop.

Cost of a BGP shortcut using next-hop MPLS

By default, next-hop MPLS is disabled. BGP uses the default BGP decision process and native IP forwarding to build BGP EMCP routes. Only IP routing tables are used to resolve routes. When next-hop MPLS is enabled, BGP uses the following configuration information to determine the cost of a BGP shortcut.

- Next-hop MPLS is enabled without an option—LSP with a fixed metric of one is used to resolve the routes.
- Next-hop MPLS is enabled with the LSP metric comparison option—BGP compares the LSP metrics and uses the metric as the IGP cost for the next hop.
- Next-hop MPLS is enabled with the **follow-igp** metrics option—BGP uses IGP metrics instead of the LSP metrics. When BGP resolves the next hop with LSP, it uses the native IGP cost for that next hop, and ignores the LSP metric of a MPLS tunnel. Then the IGP cost of each next hop is compared, and only paths with the lowest values are considered for ECMP.

LSP with a fixed metric of one when next-hop is enabled

When you enable next-hop MPLS without an option, BGP uses an LSP with a fixed metric of one to resolve the routes. For routes that cannot be resolved through the MPLS tunnels, the Extreme device uses the routing table to resolve BGP next hops.

The following steps describe the resolution process for each unique BGP next hop.

1. BGP determines when an LSP can be used to resolve the route for each unique BGP next hop. When BGP can resolve the route, it does not check the native IP routing table. For the next hops that cannot be resolved through the LSP tunnels, the Extreme device uses the IP routing table.
2. For each BGP next hop that is resolved by an LSP, then all possible LSPs to the BGP next hop are selected by default.
3. BGP internally sets this next hop's IGP cost to one instead of the true LSP metric to force it to be the preferred hop instead of a hop resolved by the native IP lookup.
4. The IGP cost is compared for each BGP next hop, and the least-value IGP cost for the next hop or hops are used to install them in the routing table.

NOTE
When the Extreme device installs a BGP route in the RIB Manager, it uses the BGP MED, not the IGP metric (IGP cost.)

LSP metric comparison when next-hop MPLS is enabled

When you enable next-hop MPLS with the LSP metric comparison option, BGP compares the configured LSP metrics instead of a fixed metric of one to resolve the routes. This option gives the flexibility to choose a native IP path over an LSP path when they have different BGP next-hops, and the native IP path has a lower IGP cost.

After BGP resolves the next hop with LSP, it uses the LSP metric as the IGP cost for this hop. Then, all of the next hops IGP costs are compared, and only the IGP cost paths with the lowest values are considered for ECMP. When any of these paths is an LSP, then only LSP paths are taken.

Follow-IGP metrics when next-hop MPLS is enabled

When you enable next-hop MPLS with the follow-IGP metrics option, BGP uses the native IGP metric for the BGP next hop resolved through IP route table, instead of the MPLS LSP metrics. BGP ignores all LSP metrics in the BGP decision process. However, the MPLS metric remains the same and is not overridden. Then the IGP cost of each next hop is compared, and only paths with the lowest values are considered for ECMP.

The advantage of using the IGP cost in a network is when this cost is significant throughout the local domain and all routing protocols, and you want to use it as a tie-breaker rather than use MPLS-specific metric value. This option works best when the MPLS LSP metrics follow the IGP cost, providing full advantage of both routing protocols and MPLS to select the best path. The advantage of doing this is that the BGP decision process on IGP cost is purely based on the IGP cost to a different next hop. MPLS tunnel is treated as a relay service.

Consider the following when enabling the use of IGP metrics instead of LSP metrics:

- Be aware and ensure that the IGP costs are consistent across the network and that you want to rely on the IGP cost to determine where to send the traffic.
- When combined with the BGP **install-igp-cost** command, you can change the route cost from BGP MED to IGP cost and is used when BGP routes are added to the RIB Manager.
- When combined with a BGP outbound policy for route **set metric-type internal** command, you can set IP over MPLS routes using IGP metric to send out as the BGP MED value.

NOTE
The previous item is primarily for BGP to set the route MED value as the IGP cost and advertise the route to a neighbor. Configuring the use of IGP metrics is based on BGP routes resolving the next hop to MPLS LSP tunnel. A route map is required to set the BGP MED value to the IGP metric by the **set metric-type internal** command.

NOTE
Mixed ECMP of LSP and native IGP paths is not supported. For a given prefix, if there are multiple BGP next hops with equal IGP cost, BGP prefers the next hop using the LSP path over the IGP path.

Configuring BGP shortcuts using next-hop MPLS

Configure BGP shortcuts using next-hop MPLS to enable BGP to use an MPLS tunnel as the preferred route to a destination network when a tunnel is available. By default, BGP considers only IP routes when building a routing table.

Before configuring next-hop MPLS, configure the LSPs.

To configure BGP shortcuts using next-hop MPLS, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable BGP routing.

```
device(config)# router bgp
```

3. Configure the autonomous system number (ASN) in which your device resides.

```
device(config-bgp-router)# local-as 10
```

4. Configure the ASN for a remote neighbor.

```
device(config-bgp-router)# neighbor 10.1.1.2 remote-as 20
```

5. Configure the ASN for a second remote neighbor.

```
device(config-bgp-router)# neighbor 10.10.1.2 remote-as 20
```

   Repeat this step for additional remote neighbors, as required.

6. Enter BGP address-family IPv4 unicast configuration mode.

```
device(config-bgp-router)# address-family ipv4 unicast
```

7. Enable next-hop MPLS with the option to compare the LSP metrics.

```
device(config-bgp-ipv4u)# next-hop-mpls compare-lsp-metric
```

   The **compare-lsp-metric** option enables BGP to compare the configured LSP metrics to determine which LSP has the lowest cost and is the preferred path. Without the option, BGP internally sets the cost of the LSPs with the lowest metric value to one instead of the actual value.

8. Exit address family configuration mode.

```
device(config-bgp-ipv4u)# Ctrl-z
```

The following configuration is an example of the previous steps and includes the configuration of the LSPs and their metrics.

```
device(config)# router mpls
device(config-router-mpls)# lsp to2
device(config-router-mpls-lsp-to2)# no enable
device(config-router-mpls-lsp-to2)# to 10.1.1.2
device(config-router-mpls-lsp-to2)# from 10.1.1.1
device(config-router-mpls-lsp-to2)# metric 10
device(config-router-mpls-lsp-to2)# enable
device(config-router-mpls-lsp-to2)#exit
device(config-mpls)# lsp to2_sec
device(config-router-mpls-lsp-to2_sec)# no enable
device(config-router-mpls-lsp-to2_sec)# to 10.10.1.2
device(config-router-mpls-lsp-to2_sec)# from 10.10.1.1
device(config-router-mpls-lsp-to2_sec)# metric 10
device(config-router-mpls-lsp-to2_sec)# enable
device(config-router-mpls-lsp-to2_sec)# exit
device(config-router-mpls)# exit
device(config)# router bgp
device(config-bgp-router)# local-as 10
device(config-bgp-router)# neighbor 10.1.1.2 remote-as 20
device(config-bgp-router)# neighbor 10.10.1.2 remote-as 20
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# next-hop-mpls compare-lsp-metric
device(config-bgp-ipv4u)# Ctrl-z
```

Verifying the next-hop MPLS LSPs

The **show mpls lsp** command verifies the configuration of the LSPs.

```
device# show mpls lsp
LSP            To              Admin Oper  Tunnel   Up/Dn Retry Active
Name           Address         State State Intf     Times Num   Path
to2            10.1.1.2        UP    UP    tnl0     1     0     2
to2_sec        10.10.1.2       UP    UP    tnl2     1     0     3
```

Enter the **show ip route** command to verify that the LSPs are in the routing table. For example:

```
device# show ip route 111.111.111.0
IP Routing Table for VRF "default-vrf"

Total number of IP routes: 170

'*' denotes best ucast next-hop

'[x/y]' denotes [preference/metric]

111.111.111.0/24

    *via DIRECT, Lsp 10, [200/0], 3m4s, iBgp, tag 0
    *via DIRECT, Lsp 12, [200/0], 3m4s, iBgp, tag 0
```

Configuring BGP shortcuts using next-hop MPLS with Follow-IGP metrics

Configure BGP shortcuts using next-hop MPLS for BGP to use IGP metrics of the next hop instead of using LSP metrics.

When combined with the BGP **install-igp-cost** command, you can change the route cost from BGP MED to IGP cost and is used when BGP routes are added to the RIB Manager.

A route map is required to set the BGP MED value to the IGP metric by the **set metric-type internal** command. When combined with a BGP outbound policy for route **set metric-type internal** command, you can set IP over MPLS routes using IGP metric to send out as the BGP MED value.

To configure BGP shortcuts using next-hop MPLS with IGP metrics, perform the following steps.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enable BGP routing.

```
device(config)# router bgp
```

3. Enter the **local-as** command to configure the autonomous system number (ASN) in which your device resides.

```
device(config-bgp-router)# local-as 10
```

4. Enter the **neighbor remote-as** command to configure the ASN for a remote neighbor.

```
device(config-bgp-router)# neighbor 10.1.1.2 remote-as 20
```

5. Enter the **neighbor remote-as** command to configure the ASN for a second remote neighbor.

```
device(config-bgp-router)# neighbor 10.10.1.2 remote-as 20
```

   Repeat this step for additional remote neighbors, as required.

6. Enter the **address-family unicast** command using the **ipv4** parameter to enter BGP address-family IPv4 unicast configuration mode.

```
device(config-bgp-router)# address-family ipv4 unicast
```

7. Enable next-hop MPLS with the IGP metric option to use IGP metrics instead of LSP metrics.

```
device(config-bgp-ipv4u)# next-hop-mpls follow-igp
```

8. Exit address family configuration mode.

```
device(config-bgp-ipv4u)# Ctrl-z
```

9. Verify the configuration.

```
device(config-bgp)# show ip bgp next-hop
```

   The **show ip bgp next-hop** command allows you to check BGP next hop resolution and the IGP cost for the next hop.

The following configuration is an example of the previous steps.

In this example, next-hop MPLS is enabled with the **follow-igp** option.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 10
device(config-bgp-router)# neighbor 10.1.1.2 remote-as 20
device(config-bgp-router)# neighbor 10.10.1.2 remote-as 20
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# next-hop-mpls follow-igp
device(config-bgp-ipv4u)# Ctrl-z
```

ECMP forwarding for IP over MPLS

ECMP hardware forwarding is supported for IP over MPLS packets when an outgoing interface is configured as a physical port and a VE interface, or configured on an MPLS tunnel. When multiple routes use ECMP to reach a destination, hardware ECMP is automatically enabled.

ECMP load sharing for IP over MPLS is supported for 2 to 64 tunnels, with a default of eight tunnels.

The hash value for each incoming packet on the route target is calculated similar to IP ECMP. The Layer 3 and Layer 4 headers are used.

QoS mapping between IP packets and MPLS

The 3-bit EXP field in the MPLS header can be used to define a Class of Service (CoS) value for packets that traverse an LSP. The CoS value specifies a priority for MPLS packets.

There are two ways that a CoS value can be applied to packets that traverse an MPLS network through an LSP:

- A CoS value is manually configured for the LSP.
- No CoS value is set for an LSP, and the Type of Service (ToS) field in the IP header is used. In this situation, the device copies the first three bits in the ToS field of the packet to the CoS (EXP) field in the MPLS header. The ToS value maps to one of the four priority queues on the device.

Configuring QoS mapping between IP packets and MPLS through an LSP

Configure CoS value for the LSP for QoS mapping between IP packet and MPLS through an LSP.

The CoS value is applied to the CoS (EXP) field in the MPLS header of all packets entering the LSP. These packets traveling through an LSP are treated with the same priority as they travel the MPLS domain.

1. In privileged EXEC mode, enter global configuration mode.

   ```
   device# configure terminal
   ```

2. Enter router MPLS mode.

   ```
   device(config)# router mpls
   ```

3. Enter LSP mode to configure the LSP tunnel.

   ```
   device(config-router-mpls)# lsp tunnel4
   ```

4. If the tunnel is enabled, disable it before changing its configuration.

   ```
   device(config-router-mpls-lsp-tunnel4)# no enable
   ```

5. Configure the CoS value for all packets traveling through the tunnel 4 LSP.

   ```
   device(config-router-mpls-lsp-tunnel4)# cos 7
   ```

   In this example, the CoS value is set to 7. You can set a value from 0 to 7. CoS does not have a default setting.

6. Enable the LSP tunnel.

   ```
   device(config-router-mpls-lsp-tunnel4)# enable
   ```

The following configuration shows the steps in the previous configuration of the CoS value in the tunnel4 LSP.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# lsp tunnel4
device(config-router-mpls-lsp-tunnel4)# no enable
device(config-router-mpls-lsp-tunnel4)# cos 7
device(config-router-mpls-lsp-tunnel4)# enable
```

IP-over-MPLS QoS TTL propagation control

The Extreme device uses a global MPLS QoS mode for IP-over-MPLS QoS TTL propagation. The mode is set to either the default uniform mode or configured pipe mode.

MPLS QoS uniform mode

By default, the device use MPLS QoS uniform mode. In the MPLS label header, the TTL field indicates the Time To Live (TTL) value for an MPLS packet. For IP-over-MPLS application at the ingress LER, an IP packet's TTL value is decremented by one and the IP checksum is recalculated. Then, the IP packet's TTL value is copied to its MPLS TTL field. At each transit LSR hop, the MPLS TTL value is decremented by one. When the MPLS TTL value reaches one or zero, the packet is discarded.

At the MPLS router that pops the label (either the penultimate LSR or the egress LER), the incoming packet's MPLS TTL value is copied to the packet's IP TTL field, the IP TTL field is decremented by one, and the checksum is re-calculated. The packet is discarded if the TTL value reaches one. The result is that each LSR in the MPLS domain is counted as one hop.

For uniform mode, the QoS mapping table maps an incoming DSCP to EXP. EXP values may change based on QoS map.

NOTE

Uniform mode is overridden for an LSP when the LSP is specifically configured a COS value. When a COS value is configured for an LSP, the device sets the tunnel as if global pipe mode is set. The outbound packet has an EXP value of the configured COS and the default system wide TTL value of 255.

MPLS QoS pipe mode

Optionally, you can configure MPLS QoS pipe mode. The ingress LER decrements the IP packet's TTL value by one and then places a value of 255 in the packet's MPLS TTL field. The MPLS TTL value is decremented by one as the MPLS packet passes through each LSR in the MPLS domain. When the label is popped, the value in the MPLS TTL field is discarded and is not copied to the packet's IP TTL field. The unlabeled IP packet's TTL is decremented by one as it passes through the egress LER. With the packet's IP TTL being decremented twice, from the time it enters the ingress LER to the time it exits the egress LER, the MPLS domain appears as two hops.

For pipe mode the TTL value is the default system wide value of 255. The default EXP value for the pipe, if the LSP is not configured, is the COS value configured in the LSP configuration. The LDP COS value is 0.

Changing the MPLS QoS mode

You can change the default MPLS QoS mode from uniform to pipe.

Perform the following steps to configure MPLS QoS pipe mode on both the ingress LER and the MPLS router that pops the label (either the penultimate LSR or the egress LER).

NOTE
If you do not configure the MPLS router that pops the label, the value in the packet's MPLS TTL field is copied into the packet's IP TTL field. This value could be as high as 255.

1. In privileged EXEC mode, enter global configuration mode.

```
device# configure terminal
```

2. Enter router MPLS mode.

```
device(config)# router mpls
```

3. Enter policy mode.

```
device(config-router-mpls)# policy
```

4. Change the MPLS QoS mode to pipe.

```
device(config-router-mpls-policy)# qos-ttl-mode pipe
```

The ingress LER places a value of 255 into the packet's MPLS TTL field, regardless of the TTL value in the packet's IP header. The packet's IP TTL value is decremented twice; once at the ingress LER and once at the egress LER. The entire MPLS domain, regardless of the number of transit LSR hops, counts as two hops for the IP TTL value.

The mode changes are applied for newly created tunnels. They are not reapplied to an existing tunnel.

5. If required, reset the default MPLS QoS mode of uniform.

```
device(config-router-mpls-policy)# no qos-ttl-mode
```

You can also use the **qos-ttl-mode uniform** command.

The following configuration shows the previous steps.

```
device# configure terminal
device(config)# router mpls
device(config-router-mpls)# policy
device(config-router-mpls-policy)# qos-ttl-mode pipe
device(config-router-mpls-policy)# no qos-ttl-mode
```

BGP or MPLS VPNs

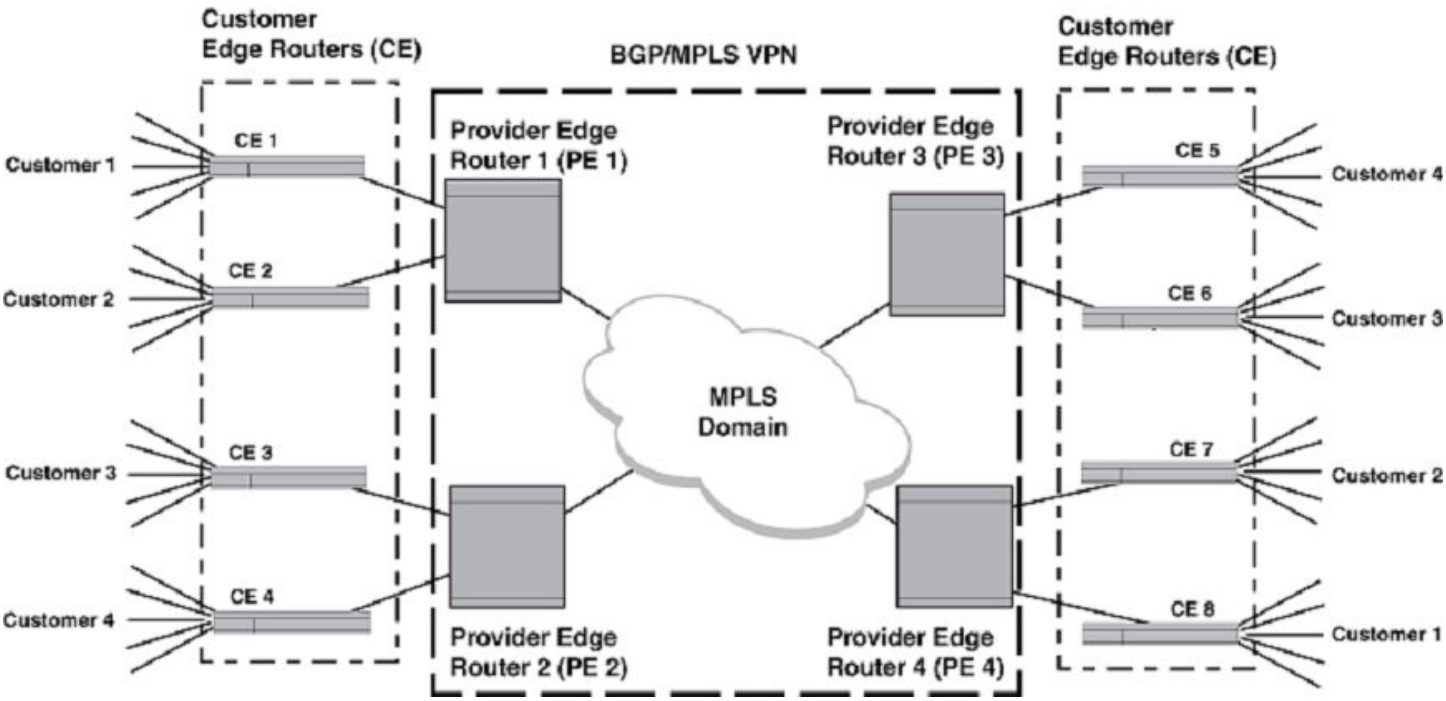
What is a BGP or MPLS VPN

The Virtual Private Connection (VPN) version 4 (v4) or version 6 (v6) feature provides connection between IPv4 or IPv6 private data network over public IPv4 network using the Multiprotocol Label Switching (MPLS) tunneling mechanism.

As defined by RFC 2547, MPLS VPN can be used by internet service providers to provide remote wide-area connectivity services using an MPLS domain for data traffic and internal Border Gateway Protocol (IBGP) to distribute routing information. By using this feature each customer network can be completely segregated from every other customer network while sharing the same infrastructure. MPLS provides scalable and efficient switching over an indeterminate group of devices along a predetermined labeled-switch-path (LSP). Using MPLS, LSPs can be set statically or determined dynamically by the Internet service providers (ISPs) to provide traffic engineering features. Border Gateway Protocol (BGP) or MPLS VPNs build on this infrastructure to provide virtual-circuit connectionless service between remote sites. Using a common MPLS-domain, multiple Virtual Private Networks (VPNs) can be configured across a service-provider MPLS core network. Each VPN provides a secure data path that allows IP packetized traffic to share the infrastructure while being effectively segregated from other VPNs that are using the same MPLS domain.

In the diagram below, four separate customers (1-4) each have remote sites. Each customer is connected to a network at a remote site through the MPLS domain while being completely segregated and secure from traffic between other sites. For instance, CE 1 and CE 8 belong to Customer 1. CE 1 is connected to the BGP or MPLS VPN network through PE 1 and CE 8 through PE 4. Using the service provider's BGP or MPLS VPN service, traffic can be forwarded between CE1 and CE8 at the same time that Customers 2 through 4 use VPNs that operate over the same network infrastructure. Different customers can even use the same IP addresses without conflicting with other customers networks or creating any routing problems.

FIGURE 31 BGP or MPLS VPN network



IETF RFC and Internet Draft support

The implementation of BGP or MPLS VPNs supports the following IETF RFCs and Internet Drafts:

BGP or MPLS VPNs

- RFC 4364: BGP or MPLS IP VPNs
- RFC 4577: OSPF as the PE or CE Protocol in BGP or MPLS IP VPNs
- RFC 4576: Using LSA Options Bit to Prevent Looping in BGP or MPLS IP VPNs (DN Bit)

BGP

- RFC 1771—A Border Gateway Protocol 4 (BGP-4)
- RFC 1997—BGP Communities Attribute
- RFC 2283—Multiprotocol Extensions for BGP-4
- RFC 2842—Capabilities Advertisement with BGP-4
- RFC 2858—Multiprotocol Extensions for BGP-4
- RFC 3107—Carrying Label Information in BGP-4
- RFC 5291 - Outbound Route Filtering Capability for BGP-4
- RFC 4360 - BGP Extended Communities Attribute

MIB spport

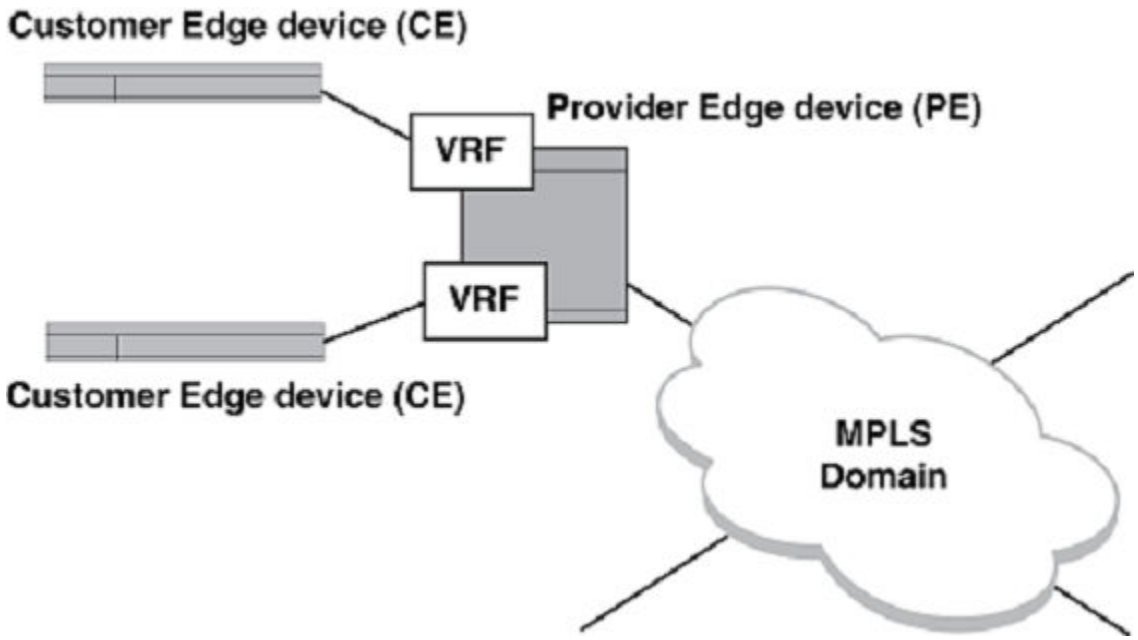
RFC 4382 - MPLS or BGP Layer 3 Virtual Private Network (VPN) Management Information Base (with full support introduced in version 03.2.00 of the Multi-Service IronWare software).

BGP or MPLS VPN components and what they do

The following components, as shown in the diagram below, comprise a BGP or MPLS VPN.

- Customer Edge device (CE)—The CE provides connectivity with a customer's network and a Provider Edge device (PE). It can advertise routes available from the customer's network using RIP, OSPF or EBGP. Alternately, the CE can create a static default route to a PE. Outbound packets from a customer's network are forwarded from the CE to the PE, and inbound packets are forwarded from the PE to the CE attached to the customer's network.
- Provider Edge device (PE)—In a BGP or MPLS VPN, the central component is the PE. The PE provides connectivity with the CE and with the MPLS domain. On one side of the PE, routing information is exchanged with the CE using either static routes, RIP, OSPF, or EBGP. On the other side, IBGP is used with BGP multiprotocol extensions to communicate with all of the other PEs that are connected to networks in the same VPN and available to the customer's network. When a CE sends packets to a PE to forward across an MPLS domain, that PE functions as an MPLS ingress Label Edge router (LER) and the PE on the other end of the domain functions as an MPLS egress LER.
- Virtual Routing and Forwarding table (VRF)—Virtual Routing and Forwarding table (VRF) - The PE maintains a Virtual Routing and Forwarding table (VRF) for each customer that is attached to it through a CE. The VRF contains routes between the PE and the CE and *Label Switched Paths (LSPs)* across the MPLS domain for each PE that is a member of the customer's VPN. VRFs are defined on interfaces of the PEs.
- Provider MPLS domain—The Provider MPLS domain is composed of Provider (P) devices. An MPLS domain can traverse more than one service provider's MPLS network. The P devices do not store any VPN information; they just switch traffic from the ingress PE device along the LSP to the egress PE device.

FIGURE 32 BGP or MPLS VPN components



BGP or MPLS VPN operation

The purpose of a BGP or MPLS VPN is to forward packets between remote sites of a customer's network through a service provider's MPLS infrastructure. The section titled BGP or MPLS VPN components and what they do on page 245 describes the network components required to perform that task. The following sections describe how those components work together to create this service:

- Creating routes in a BGP or MPLS VPN on page 246
- Routing a packet through a BGP or MPLS VPN on page 247

Creating routes in a BGP or MPLS VPN

The diagram below illustrates the various components involved in creating routes in a BGP or MPLS VPN.

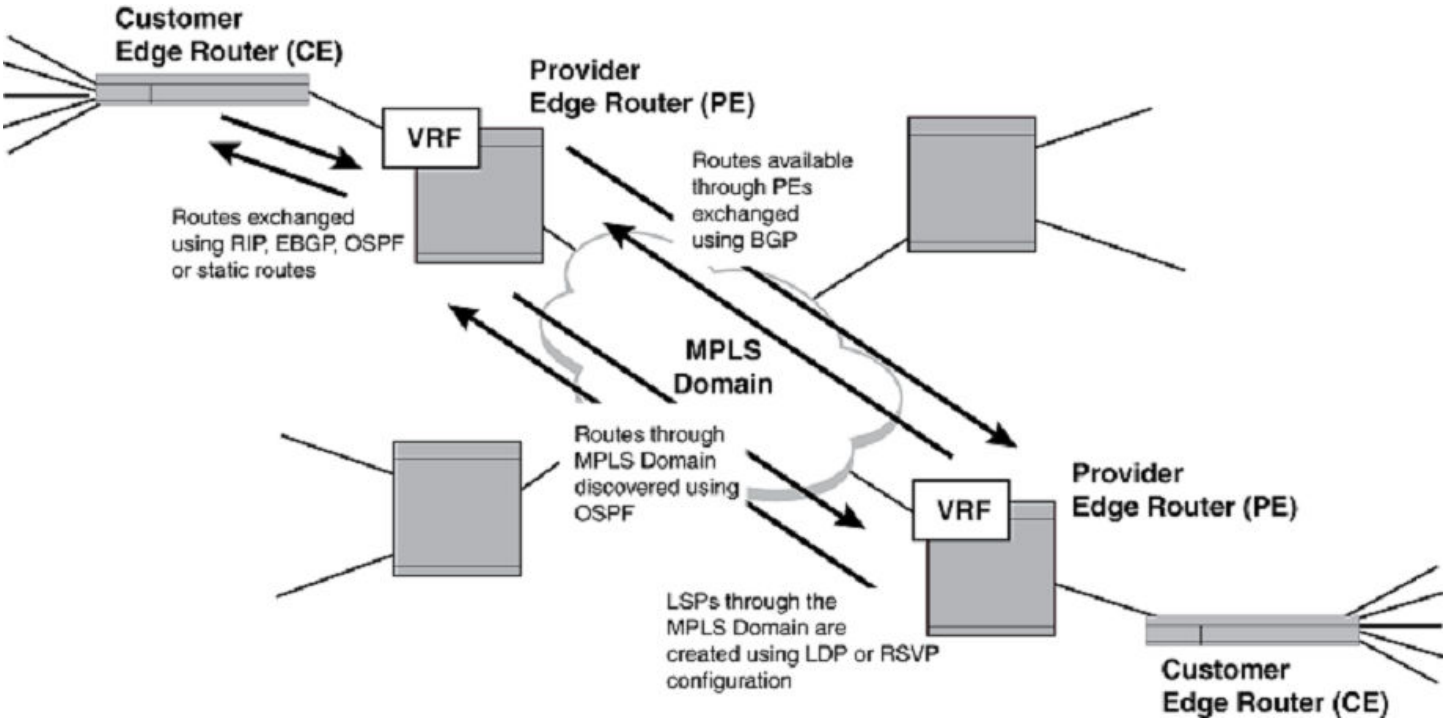
A CE device maintains the connection to the customer's network and is configured within that network to share access to its available network prefixes and to receive packets from other VPN-connected networks. That CE is connected to a PE through an interface that is configured for a specified VRF for connection to the BGP or MPLS VPN. This connection places the CE in the BGP or MPLS VPN. Routes that are available through the CE are then made available to the PE using , OSPF, EBGP or a static route. These routes are then stored in the VRF where they are associated with the VPN. The route from the CE to the PE is kept in the CEs routing table.

The PE device is connected to the MPLS domain through one or more interfaces. The PE must advertise the routes that it has available in it's VRF tables across the MPLS domain to its PE peers. Available routes in the VRF are prepended with a Route Distinguisher (RD) and advertised across the MPLS domain using IBGP. The PEs can either be configured for IBGP as either full mesh or with a route reflector to allow greater scalability. Routes that are advertised from other PEs in the VPN are received at the PE and collected in the VRF table. This procedure establishes which other PEs are in the VPN and what networks are available through them.

OSPF or ISIS is used as the Interior Gateway Protocol (IGP) within the service provider's MPLS domain to provide connectivity. OSPF or ISIS also populates the traffic engineering (TE) used by RSVP-TE.

Labeled Switch Paths (LSPs) are then created using Label Distribution Protocol (LDP), Resource Reservation Protocol (RSVP) configurations in the MPLS domain. Using this protocol, the PE obtains an LSP required to switch traffic to the other PEs. The network is now populated with all of the routes required to forward packets between the customer's networks.

FIGURE 33 BGP or MPLS VPN route discovery

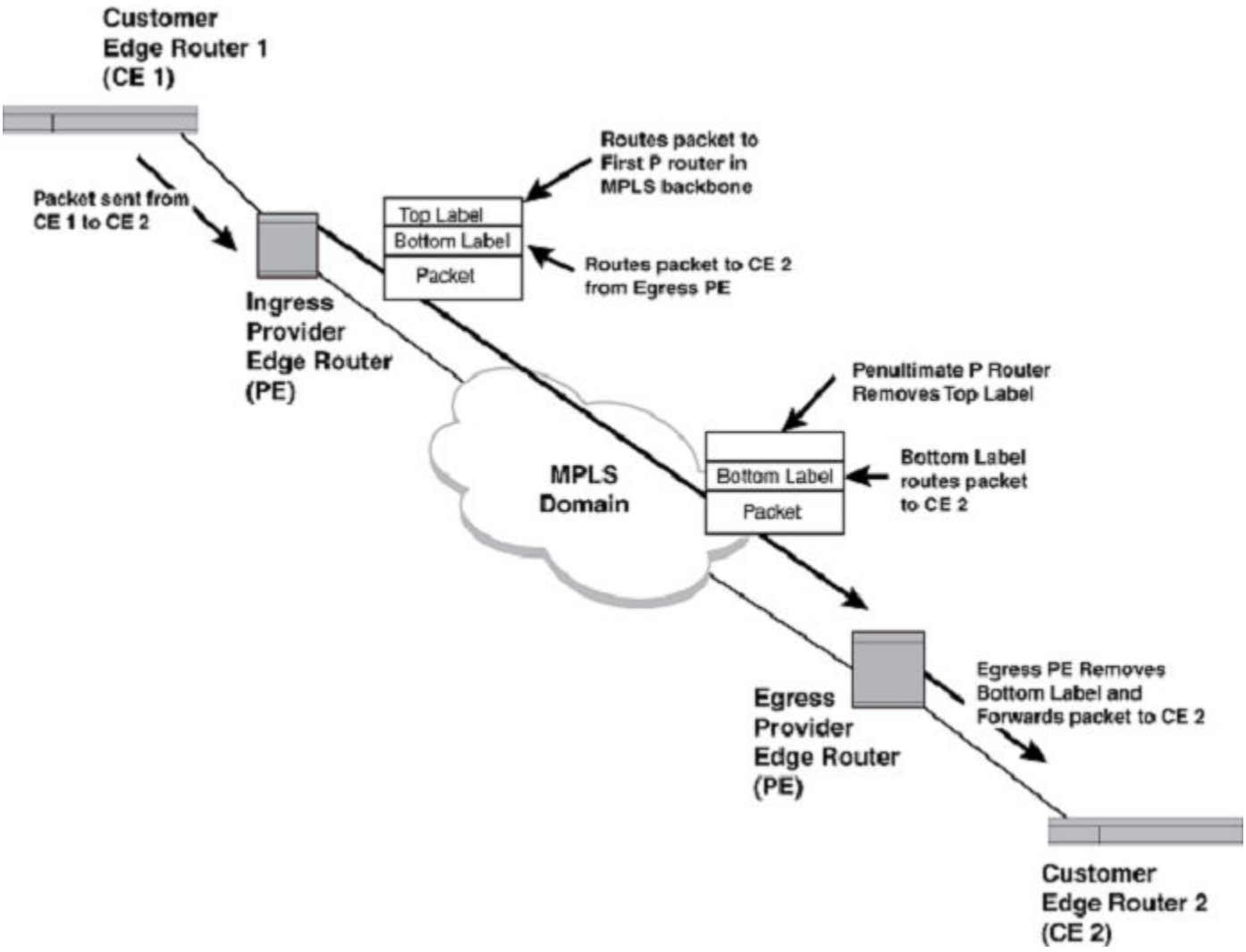


Routing a packet through a BGP or MPLS VPN

When a packet is forwarded from a CE to a PE, a bottom label is attached to the packet by the PE that is associated with the final destination. This label is obtained from the egress PE as part of the route discovery conducted by IBGP. Then, the top label which is obtained by the LSP connecting to the egress PE is added to the packet. The packet is then forwarded through the MPLS domain and is switched using the top label. At the penultimate device in the LSP, the top label is removed and the packet is forwarded to the egress PE. The egress PE uses the inner label to identify the CE to which the packet must be forwarded. The egress PE removes the inner label and forwards the packet to the correct CE.

The diagram below describes how a packet is forwarded through a BGP or MPLS VPN.

FIGURE 34 Routing a packet through a BGP or MPLS VPN



L3VPN over MPLS tunnel

This feature provides a mechanism to connect private IPV4 and IPV6 data networks over a public IPV4 network using MPLS tunnel mechanism.

L3VPN encapsulation at ingress node

L3 packets are encapsulated with L3VPN label and are sent over MPLS tunnel in the ingress node. In the L3VPN ingress node, VRF is identified from the incoming L3 interface. Packets undergo route lookup to identify the route to forward the packet. Based on the route information, outgoing L3VPN label is decided. Out Label information is obtained as part of the BGP route exchange.

NOTE

ECMP for L3VPN is supported along with other native property of underlying MPLS tunnel.

Based on the underlying MPLS tunnel, outgoing packets could be in any of the following formats.

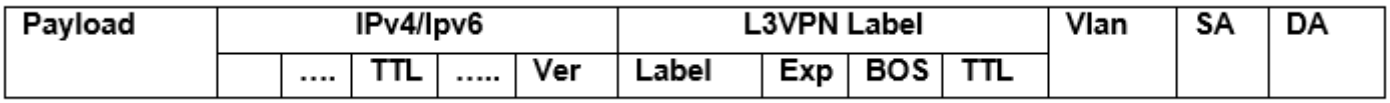
- L2Hdr + L3VPN Label + IP Payload (single hop tunnel)
- L2Hdr + MPLS Tunnel Label + L3VPN Label + IP Payload (multi hop MPLS tunnel)
- L2Hdr + By-Pass Lbl + MPLS Tunnel Label + L3VPN Label + IP Payload (multi hop tunnel over a bypass)

Extreme devices support uniform and pipe mode. Short-pipe mode is not supported. For single hop MPLS tunnels, in Pipe mode, QoS parameters are propagated to MPLS header.

L3VPN label termination at egress node

In Layer 3 VPN, tunnel termination occurs at egress node.

FIGURE 35 L3 VPN packet format



L3 VPN packets at egress node come with the header that must have L3VPN label (MPLS), and the DA Mac must be the incoming interface (physical or Virtual Ethernet) MAC address.

On egress node, the L3VPN label is terminated, and the VRF-id is be derived to initiate the IP lookup with the VRF-ID and in case of matching DIP entry, traffic forwarding is processed.

Incoming packets on egress node are processed in different ways depending on different modes configured (RFC 3270) on the device. Extreme devices support uniform and pipe mode. Short-pipe mode is not supported.

Packets with L3VPN label TTL=1 and TTL=0 are trapped to CPU and they are dropped. If a tunnel termination occurs, the packet size is reduced. If the outgoing port MTU configured size is lesser than this outgoing packet size, packets are sent to CPU for fragmentation depending on DF bit setting. Extreme SLX-OS supports tunnel-termination statistics per VPN label

Tunnel termination happens at egress node. The L3VPN packet at egress node comes withL3VPN label (MPLS) and the DA Mac is the incoming interface MAC address. On egress node, the L3VPN label is terminated and the vrf-id is derived from the label value. After the label termination, IP lookup is launched with the derived vrf-id and in case of matching DIP entry, traffic forwarding happens. The outgoing packet from this node is the regular L3 packet.

> NOTE
> Currently, support is only for PHP. MPLS Tunnel Label is terminated at PHP node. Egress PE will always receive packet with only L3VPN label.

After the L3VPN label termination, IP lookup is launched based on packet header's next nibble field after the L3VPN Label. If it is 4, IPv4 route lookup is launched. If it is 6, IPv6 lookup is launched.

> NOTE
> IPv4/IPV6 lookup is not dependent on VRF address-family configuration. If DA MAC is not MyMAC (incoming interface MAC), regular L2 flooding happens.

Configuring BGP or MPLS VPNs on a PE

Configuring BGP or MPLS VPNs on a PE involves the following configuration at a high level:

- Configuring VRF
- Associating the VRF to the interface
- Configuring BGP under VPNV4 unicast
- Configuring MPLS using LDP or RSVP

Defining a VRF routing instance

A single PE can contain one or more VRFs. Each of these VRFs must be defined separately on a PE. A PE distributes routes and route packets to other members of the same VRF but not to other VRFs. The VRF name can be any string that the user wants to define it as.

To define the VRF routing instance VPN1 on a PE, enter the following command.

```
config)# vrf VPN1
(config-vrf-vpn1)# exit-vrf
(config)#
```

Syntax: [**no**] **vrf** *vrf_name*

Configures a VRF table on the device with the name *vrf_name* and puts the device in config-vrf mode.

The *vrf_name* parameter specifies a name for the VRF being created.

Syntax: [no] exit-vrf

The **exit-vrf** command moves the user out of the VRF configuration mode for the VRF the user is configuring.

Assigning a Route Distinguisher to a VRF

Each instance of a VRF must have a unique Route Distinguisher (RD) assigned to it. The RD is pre-pended on any address being routed or advertised. The RD can be defined as either ASN-relative or IP address-relative. Because the RD is unique to an instance of a VRF, it allows the same IP address to be used in different VPNs without creating any conflict.

To assign a Route Distinguisher (RD) for a VRF based on the AS number 3 and the arbitrary identification number 6, enter the following command.

```
(config-vrf)# rd 3:6
```

Defining IPv4 or IPv6 address families of a VRF

Each address family configuration level allows the user to access commands that apply to that particular address family only.

To define IPv4 or IPv6 address families of a VRF, enter the following command.

```
device(config)# vrf VPN1
device(config-vrf-vpn1)# address-family ipv4 unicast
device(config-vrf-vpn1-ipv4)# exit-address-family
device(config-vrf-vpn1)# exit-vrf
```

Defining automatic route filtering

Each VRF is configured with import and export route targets. The export route target sets an extended community attribute number that is appended to all routes that are exported from the VRF. The import route target value sets a filter that determines the routes that are accepted into the VRF. Any route with a value in its import route-target contained in its extended attributes field matching the value in the VRFs import route target is accepted. Otherwise, the route is rejected. This process is referred to as automatic route filtering.

To define an import route target of 3:6 and an export route target of 3:8 for a VPN, enter the following commands.

```
        device(config-vrf)# route-target import 3:6
device(config-vrf)# route-target export 3:8
```

Syntax: [no] route-target [**import** | **export** | **both**] *route-target*

This command associates a route target specified by the route-target variable with a specified VRF for control on routes.

The **import** parameter specifies that routes with route-target extended community attributes matching the specified route-target variable can be imported into the VRF where this command is configured.

The **export** parameter specifies the route-target extended community attributes that are attached to routes export from the specified VRF.

The **both** parameter specifies that both the import and export values apply to the specified route-target variable for the VRF where this command is configured. This is the default state. It applies when no specific value for this parameter is set.

The *route-target* variable specifies a target VRF extended community. Like a route distinguisher, it is either AS-relative or IP address-relative.

Assigning a VRF routing instance to an interface

Once a VRF routing instance is defined, it must be assigned to one or more virtual or physical interfaces on a PE.

To assign the VRF named VPN1 to Ethernet interface 1/1, enter the following commands.

```
        device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# vrf forwarding VPN1
```

Syntax: [no] vrf forwarding *vrf-name*

The *vrf-name* variable is the name of the VPN that the interface is being assigned to.

Setting up cooperative route filtering

Automatic route filtering in VRFs is provided through the **route-target** import command. By placing this command in the VRF configuration, routes can be filtered from being imported into a given VRF. Routes with extended community route targets matching the VRF's import route-targets are permitted into a VRF. Otherwise, the routes are rejected.

The cooperative route filtering feature requires that the user sets a send command on the device that is sending the ORF, and a receive command on the device that is installing the ORF. To configure the sending device, use the following command in the VPNv4 address family or VPNv6 address family.

```
device(config-bgp-vpnv4u)# neighbor 10.3.3.1 capability orf extended-community send-vrf-filter
```

Syntax: [no] neighbor *neighbor_IPaddress* **capability orf extended-community send-vrf-filter**

To configure the peering device use the following command in the VPNv4 address family or VPNv6 address family.

```
(config-bgp-vpnv4u)# neighbor 10.3.3.2 capability orf extended-community receive
```

Importing and exporting route maps

Route-maps configured using the **route-map** command can be applied to a VRF to provide filtering of VPNv4 or VPNv6 routes between PEs in a BGP or MPLS VPN. When a route-map is applied to a VRF, only VPNv4 routes are filtered. Other routes such as static routes, connected routes, OSPF VRF routes, or BGP CE side routes are not affected. Because the route map is applied to the VRF, it filters traffic to all connected PEs. This is in contrast to applying a route-map using the BGP neighbor. In that case, the route map applies to routes imported from or exported to the neighbor that is specified.

Route maps applied to a VRF can coexist with route maps that are applied to a BGP neighbor. The user can filter routes from being imported into a VRF using the import and export route commands. This allows the user to accept or deny the routes for one VRF without affecting the routes that are imported or exported from other VRFs. To do this, the user must define a route-map import or export command.

To configure a VRF to apply the import route map ImportOne, use the following command at the VPNv4 prompt.

```
(config)# vrf vrfone
(config-vrf-vrfone)# import map ImportOne
(config-vrf-vrfone)# exit-vrf
(config)#
```

To configure a VRF to apply the export route map ExportOne, use the following command at the VPNv4 prompt.

```
config)# vrf vrfone
(config-vrf-vrfone)# export map ExportOne
(config-vrf-vrfone)# exit-vrf
(config)#
```

Defining an extended community for use with a route map

Routes can be filtered in or out of a PE by the use of an IP extended community to identify them. In this situation, a route is identified by its extended community variable. It is entered as a route target in an IP extended community list and then matched in a route-map command. This route map is then applied from the PE that is defining the route to be filtered to the PE where the route filter is to be implemented by using a **neighbor route-map** command. When a VRF exists on the neighbor that exports the route-target being blocked, all routes from that VRF are blocked from being sent to the PE where the filter is defined.

To define the IP extended community list 20 to define route target RT 100:6 to be denied, enter the following command.

```
(config)# ip extcommunity-list 20 deny rt 100:6
```

Creating a VPNv4 route reflector

PE devices in a BGP or MPLS VPN share routes between each other using IBGP. This can be accomplished using a full mesh configuration or a route reflector can be used to simplify a networks topology and improve scalability. While the general concepts are the same for using Route Reflectors in a normal IBGP network as in an BGP or MPLS VPN, there are some differences. In addition, there are special conditions that apply when a route reflector is configured for normal IPv4 BGP traffic (IPv4) and for BGP or MPLS VPN traffic (VPNv4). The differences and special considerations are described in the following:

Special considerations when configuring a route reflector for both IPv4 and VPNv4:

- A VPNv4 route does not need to be installed in any VRF before being reflected.
- Route reflector configurations for IPv4 and VPNv4 are separated in different address family configurations.

- For a VPNv4 route installed to a VRF, the reflected VPNv4 route still carries the original RD and PA.
- When there is a route reflector configuration change, a warning message is displayed that requests the user to clear the neighbor session.

Specific commands for VPNv4 - There are VPNv4 specific commands that must be configured to configure a route reflector for a BGP or MPLS VPN under address family VPNv4. A route reflector can be configured on a PE for IPv4 and VPNv4 or for either exclusively. When the user is configuring a route reflector for a BGP or MPLS VPN, the user must configure it specifically using the VPNv4 specific commands.

To create a VPNv4 route reflector with a client at the IP address 10.11.11.2, enter the following commands at the VPNv4 level of BGP Config level.

```
(config-bgp-vpnv4u)# neighbor 10.11.11.2 route-reflector-client
```

NOTE
You must follow the same considerations and create a VPNv6 route reflector also.

Configuring autonomous system number override

There are some situations where a customer wants to connect to a service provider's BGP or MPLS VPN network using the same AS number at more than one site. This can create a problem because it is the default BGP procedure to reject routes from the same AS. One solution to this problem is to configure a PE router to override the AS_PATH attribute of its BGP neighbor. This is accomplished by configuring the **neighbor as-override** command on the PE. When this is enabled, the PE device determines when the AS_PATH attribute in a route intended for a neighbor CE contains the same AS number as the CE. When this is determined, the PE device substitutes its own AS number for the CEs in the AS_PATH attribute. The CE is then able to receive the route. The following additional conditions apply when this feature is in effect:

- In a situation where the AS_PATH attribute contains more than one occurrence of the CEs AS number in the initial sequence, the PE device replaces all those occurrences with its own AS number.
- The PE device adds its own AS number to the AS_PATH attribute just as it would normally.

The following command configures the PE device to replace its attached CEs AS number with its own AS number. BGP neighbor at IP address 10.33.36.2 the configuration of PE 2 required to enable Autonomous System number override for the BGP neighbor CE 2.

To configure a PE device to replace its attached CEs AS number with its own AS number, enter the following commands at the VRF level of the BGP Config level.

```
(config-bgp-vpnv4u)# neighbor 10.33.36.2 as-override
```

Configuring a PE to allow routes with its AS number

BGP rejects routes that contain its own AS number within its AS_PATH attribute to prevent routing loops. In an MPLS or VPN hub and spoke topology this can stop legitimate routes from being accepted. The **allowas-in** command fixes this problem by allowing the user to set a parameter that disables the AS_PATH check function for routes learned from a specified location.

To configure a PE to disable the AS_PATH check function for routes sent to it by its BGP neighbor (a CE device with the IP address 10.33.36.2) for a maximum limit of three occurrences of the route, enter the following command at the BGP VRF configuration level.

```
(config-bgp-ipv4u-vrf)# neighbor 10.33.36.2 allowas-in 3
```

Syntax: [no] neighbor *IPaddress* **allowas-in** *asn_limit*

The *IPaddress* is the IP address of the neighbor CE device from which the PE device can accept routes that have the same AS number.

The asn_limit value prevents loops by limiting the number of occurrences that the PEs AS number can be accepted in routes that are received from the specified device.

Setting up LSPs per VRF

IBGP is used between PEs to determine routes that are available between VRFs. These routes are linked to a Label Switched Path (LSP) that has been defined separately either as a static path or using LDP or RSVP. The LSP is used to tunnel through the MPLS domain to the destination PE. Under most circumstances, the default route between two PEs is chosen by IBGP between the VRFs with the PEs loopback address as the next hop. When there is a single loopback on the PE, the same LSP tunnel is the only path used between any VRF defined on a PE and VRFs on other specified PEs.

More than one LSP can be configured between PEs however, where each LSP is associated with a different Loopback address on the PE. In this case, any loopback address on a PE can be assigned as the nexthop address for a specific or multiple VRFs. This allows the user to assign some VRFs on a PE to one LSP and other VRFs to a different LSP. Through this method, traffic from different VRFs can be assigned to LSPs that provide different qualities of service. This feature can also be employed to provide for load-balancing across the MPLS domain.

To configure a PE device to use different LSPs, a BGP next hop must be configured for a VRF as the following example illustrates.

```
(config)# vrf blue
(config-vrf-blue)# bgp next-hop loopback 2
(config-vrf-blue)# exit-vrf
(config)#
```

Syntax: [no] bgp next-hop *loopback-interface*

The *loopback-interface* variable is the number of the loopback interface that the user is assigning to the VRF as a BGP next hop. The loopback address becomes the defined VRF's nexthop for its VPNv4/VPNv6 routes that are sourced by this device only when:

- The loopback interface exists and has an IP address set.
- The loopback interface has an IP a subnet mask of /32
- The loopback interface is in the default VRF.

When these conditions are not met, the default nexthop is used.

For a detailed example of this feature refer to Setting an LSP for each VRF on a PE on page 308.

Configuring OSPF sham links

OSPF can be used to propagate links between a Customer Edge device (CE) and a Provider Edge device (PE). Normal operation of this type of network assumes that the only connections between CEs pass through the provider network. However, when other links or routes between the CEs exist within the same area, problems can arise due to the OSPF preference for Intra-area links over Inter-area links.

Problems can be avoided by creating a virtual intra-area OSPF link between two PEs. This virtual link is called a sham link. When the OSPF instances exist in the same area, a sham link causes OSPF to treat the route through the service provider network as an intra-area link instead of an inter-area link.

> NOTE
> When no backdoor link exists, no purpose exists for creating a sham link.

A cost is assigned to the sham link to help the OSPF network determine when to route over the sham link and when to route over the backdoor link. Because this virtual link (sham-link) appears as an intra-area link, the OSPF areas in which each of the PEs reside must be the same.

To configure an OSPF sham link, use the command for creating a sham link on both the local device and the remote PE device. Before attempting to create a sham link, note the following important information:

- For sham links to work, OSPF cannot be configured on the loopback interface in the applicable area.
- The redistribution of BGP to OSPF must be configured.
- A BGP VPN4 route to the loopback address must exist in both of the pertinent VRFs' routing tables.
- After the BGP VPN4 route exists in the VRF IP route table, the hello (and other) packet exchanges can go through for sham links even when the backdoor CE link does not exist.

The first example that follows illustrates the command for creating an OSPF sham link between PE devices. The command shows the command entry on one device with a source IP address of 10.2.2.1 and destination address of 10.2.2.2. The second example shows the complete configuration sequence (from both PE devices) and uses the **show ip route vrf** command that is used for viewing the sham link.

Use this command in the OSPF VRF configuration level.

```
(config-ospf-router)# area 1 sham-link 10.2.2.1 10.2.2.2 cost 10
```

Syntax: [no] area *area_id* **sham-link** *source_address /destination_address* **cost** *cost_value*

Possible values :

The *area_id* variable is the ID number of the OSPF area assigned to the sham ink being defined in this command.

The *source_address* variable is the IP address of the source PE device.

The *destination_address* variable is the IP address of the destination PE device.

The *cost_value* variable sets the OSPF cost for sending packets over the sham link. This parameter can be a numeric value in the range 1 - 65535.

Sham link configuration on PE1

The following illustrates the configuration for PE1:

```
router ospf vrf CustomerA
area 1
area 1 sham-link 172.31.255.1 172.31.255.2 cost 1
redistribution bgp

interface loopback 2
vrf forwarding CustomerA
ip address 172.31.255.1/32
!
```

Sham link configuration on PE2

The following illustrates the configuration for PE2:

```
router ospf vrf CustomerA
area 1
area 1 sham-link 172.31.255.2 172.31.255.1 cost 1
redistribution bgp

interface loopback 2
vrf forwarding CustomerA
ip address 172.31.255.2/32
!
```

Configuring OSPF on a PE device to redistribute BGP-VPNv4 or VPNv6 routes

To allow OSPF route exchange between a specified VRF on a PE device and its associated CE device, OSPF must be configured to redistribute BGP routes from the local AS as described in the following steps:

Defining an OSPF instance in a VRF

To define an OSPF instance in VRF VPN1, enter the following command at the OSPF configuration level.

```
device(config)# router ospf vrf VPN1
```

Creating an OSPF area in an OSPF VRF instance

To create OSPF area 1 in OSPF VRF instance VPN1, enter the following command in the OSPF VRF Instance configuration level.

```
device(config-ospf-router)# area 1
```

Creating a domain identifier in an OSPF VRF instance

To create OSPF domain identifier 10.0.0.100 in OSPF VRF instance VPN1, enter the following command in the OSPF VRF Instance configuration level.

```
device(config-ospf-router)# domain-id 10.0.0.100
```

Assigning a domain tag in an OSPF VRF instance

To assign OSPF domain tag 1200 in OSPF VRF instance VPN1, enter the following command in the OSPF VRF Instance configuration level.

```
device(config-ospf-router)# domain-tag 1200
```

The *domain_tag* parameter specifies an arbitrary four-byte quantity. It is added in tag fields of Type-5 and Type-7 LSAs generated by a PE device for redistributed BGP-VPNv4 routes.

When not specified, the domain-tag value is calculated from the autonomous system number of the MPLS domain.

Ping and Traceroute for layer-3 VPNs

The Ping and Traceroute utilities have been enhanced to help with management of Layer-3 VPNs.

Ping VRF

There is a VRF option for the **ping** command. To use this option, enter the following command.

```
device# ping vrf blue
```

Syntax: ping vrf { *vrf-name* | *ip-address* }

The *vrf-name* is the name of the VRF that the user wants to send a ping packet to.

The *ip-address* is the IP address containing the VRF to which the user wants to send a ping packet.

Traceroute VRF

The is a VRF option for the **traceroute** command. To use this option, enter the following command.

```
device# traceroute vrf 10.10.10.10
```

Syntax: traceroute vrf { *vrf-name* | *ip-address* }

The *vrf-name* is the name of the VRF that the user wants to conduct a traceroute to.

The *ip-address* is the IP address containing the VRF to which the user wants to conduct a traceroute.

Displaying BGP or MPLS VPNv4 or VPNv6 information

Use the **show** commands on both VPNv4 and VPNv6 to display the information for VPNv4 and VPNv6 respectively.

Displaying VPNv4/VPNv6 route information

The user can display route information about VPNv4/VPNv6 routes by entering the following command at any level of the CLI.

```
        device# show ip bgp vpnv4
Total number of BGP VPNv4 Routes: 285
Status codes: s suppressed, d damped, h history, * valid, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network            Next Hop             Metric LocPrf Weight      Path
Route Distinguisher: 1:1
*i  10.80.1.1/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.2/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.3/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.4/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.5/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.6/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.7/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.8/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.9/32       10.2.2.2         100    0      206 311 i
*i  10.80.1.10/32      10.2.2.2         100    0      206 311 i
*i  10.80.1.11/32      10.2.2.2         100    0      206 311 i
*i  10.80.1.12/32      10.2.2.2         100    0      206 311 i
*i  10.80.1.13/32      10.2.2.2         100    0      206 311 i
*i  10.80.1.14/32      10.2.2.2         100    0      206 311 i
*i  10.80.1.15/32      10.2.2.2         100    0      206 311 i
*i  10.80.1.16/32      10.2.2.2         100    0      206 311 i
*i  10.80.1.17/32      10.2.2.2         100    0      206 311 i
*i  10.80.1.18/32      10.2.2.2         100    0      206 311 i
--More--, next page: Space, next line: Return key, quit: Control-c
```

This display shows the following information.

TABLE 10 BGP4 summary information

This field...	Displays...
Total number of BGP VPNv4 Routes:	The number of BGP VPNv4 routes.
Status or Status Codes	The route's status, which can be one or more of the following: • A - AGGREGATE. The route is an aggregate route for multiple networks. • B - BEST. BGP4 has determined that this is the optimal route to the destination.

TABLE 10 BGP4 summary information (continued)

This field...	Displays...
	NOTE When the "b" is shown in lowercase, the software was not able to install the route in the IP route table. • b - NOT-INSTALLED-BEST. The routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table due to the rib-route-limit (or RTM route table size limit) and always-propagate option to allow the propagating those best BGP routes. • C - CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation. • D - DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable. • H - HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now. • I - INTERNAL. The route was learned through BGP4. • L - LOCAL. The route originated on this device. • M - MULTIPATH. BGP4 load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B". NOTE When the "m" is shown in lowercase, the software was not able to install the route in the IP route table. • S - SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors. NOTE This field appears only when the user enters the **route** option.
Origin code	A character the display uses to indicate the route's origin. The origin code appears to the right of the AS path (Path field). The origin codes are described in the command's output.
Route Distinguisher	A unique ID that is prepended on any address being routed or advertised from a VRF. The RD can be defined as either ASN-relative or IP address-relative as described: • ASN-relative - Composed of the local ASN number followed by a ":" and a unique arbitrary number. For example: 3:6 • IP address-relative - Composed of the local IP address followed by a ":" and a unique arbitrary number.
Network	IP address or mask of the destination network of the route.
Next Hop	The next-hop device for reaching the network from this device.
Metric	The value of the route's MED attribute. When the route does not have a metric, this field is blank.
LocPrf	The degree of preference for this route relative to other routes in the local AS. When the BGP4 algorithm compares route on the basis of local preference, the route with the higher local preference is chosen. The preference can have a value from 0-4294967295
Weight	The value that this route associates with routes from a specific neighbor. For example, when the device receives routes to the same destination

TABLE 10 BGP4 summary information (continued)

This field...	Displays...
	from two BGP4 neighbors, the device prefers the route from the neighbor with the larger weight.
Path	The routes AS path.

To clear the VPNv4 routing table, the user must enter the following commands.

```
        device# clear ip bgp vpnv4 neighbor all soft out
device# clear ip bgp vpnv4 neighbor all soft in
```

Syntax: clear ip bgp vpnv4 { **dampening** | **flap-statistics** | **neighbor** }

The **dampening** parameter clears route flap dampening information.

The **flap-statistics** parameter clears route flap statistics.

The **neighbor** parameter clears BGP neighbors.

Displaying VPNv4/VPNv6 route information for a specified IP address

To display only the routes to a specified network, enter a command such as the following at any level of the CLI.

```
        device# show ip bgp vpnv4 10.2.2.0/24
Route Distinguisher: 2:1
Number of BGP Routes matching display condition : 1
Status codes: s suppressed, d damped, h history, * valid, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network                         Next Hop               Metric      LocPrf      Weight      Path
*i  10.2.2.0/24                            10.4.4.4        1           100     0       ?
```

Syntax: show ip bgp vpnv4 *ip-address*/*mask*

The *ip-address*/*mask* parameter specifies a particular route. When the user also uses the optional **longer-prefixes** parameter, then all statistics for routes that match the specified route or have a longer prefix than the specified route are displayed. For example, when the user specifies **10.157.0.0 longer** , then all routes with the prefix 10.157 or that have a longer prefix (such as 10.157.22) are displayed.

The number of BGP routes matching display conditions field in this display is described in the table below.

TABLE 11 Route flap dampening statistics

This field...	Displays...
Number of BGP Routes matching display conditions	The number of routes to the network specified as a parameter in the **show ip bgp vpnv4***ip-addr* command.

Displaying VPNv4/VPNv6 attribute entries information

The route-attribute entries table lists the sets of BGP VPNv4/VPNv6 attributes stored in the device memory. Each set of attributes is unique and can be associated with one or more routes. In fact, the device typically has fewer route attribute entries than routes. To display the route-attribute entries table at any level of the CLI.

```
      device# show ip bgp vpn attribute-entries
        Total number of BGP Attribute Entries: 55
1       Next Hop  :0.0.0.0            Metric   :0              Origin:IGP
        Originator:0.0.0.0            Cluster List:None
        Aggregator:AS Number :0       Router-ID:0.0.0.0       Atomic:None
        Local Pref:100                Communities:Internet
        Extended Community: RT 600:1
        AS Path   :310
```

```
        Address: 0x24644060  Hash:45 (0x0100036e) Reference Counts: 0:0:30
2       Next Hop  :0.0.0.0            Metric   :0              Origin:IGP
        Originator:0.0.0.0            Cluster List:None
        Aggregator:AS Number :0      Router-ID:0.0.0.0         Atomic:None
        Local Pref:100               Communities:Internet
        Extended Community: RT 600:1
        AS Path   :311
        Address: 0x24645f48  Hash:47 (0x01000370) Reference Counts: 0:0:30
3       Next Hop  :2.2.2.2            Metric   :0              Origin:IGP
        Originator:0.0.0.0            Cluster List:None
        Aggregator:AS Number :0      Router-ID:0.0.0.0         Atomic:None
        Local Pref:100               Communities:Internet
        Extended Community: RT 100:1 RT 200:1
        AS Path   :206 311
        Address: 0x24645538  Hash:276 (0x0100087a) Reference Counts: 30:0:0
```

This display shows the following information.

TABLE 12 BGP VPNv4 / VPNv6 attribute entries

This field...	Displays...
Total number of BGP Attribute Entries	The number of routes contained in the BGP4 route table for this device.
Next Hop	The IP address of the next hop device for routes that have this set of attributes.
Metric	The cost of the routes that have this set of attributes.
Origin	The source of the route information. The origin can be one of the following: • **EGP** - The routes with this set of attributes came to BGP through EGP. • **IGP** - The routes with this set of attributes came to BGP through IGP. • **INCOMPLETE** - The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPF or RIP. When BGP4 compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE.
Originator	The originator of the route in a route reflector environment.
Cluster List	The route-reflector clusters through which this set of attributes has passed.
Aggregator	Aggregator information: • **AS Number** shows the AS in which the network information in the attribute set was aggregated. This value applies only to aggregated routes and is otherwise 0. **Router-ID** shows the device that originated this aggregator
Atomic	Whether the network information in this set of attributes has been aggregated and this aggregation has resulted in information loss. • TRUE - Indicates information loss has occurred • FALSE - Indicates no information loss has occurred **NOTE** Information loss under these circumstances is a normal part of BGP4 and does not indicate an error.
Local Pref	The degree of preference for routes that use this set of attributes relative to other routes in the local AS.
Communities	The communities that routes with this set of attributes are in.

TABLE 12 BGP VPNv4 / VPNv6 attribute entries (continued)

This field...	Displays...
Extended Community	The extended community attributes.
AS Path	The ASs through which routes with this set of attributes have passed. The local AS is shown in parentheses.
Address	This is an internal value used for debugging purposes only.
Hash	This is an internal value used for debugging purposes only.
Reference Counts	This is an internal value used for debugging purposes only.

Displaying VPNv4/VPNv6 filtered routes information

To view BGP VPNv4 filtered paths information, enter the following command.

```
device# show ip bgp vpnv4 filtered-routes
```

Displaying VPNv4/VPNv6 route distinguisher information

In order to view the BGP VPNv4 information for routes that contain a specific route distinguisher, enter the following command.

```
       device# show ip bgp vpnv4 rd 5:1 detail
Total number of BGP Routes: 34
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
1       Prefix: 10.6.1.0/24,  Status: I,  Age: 16h9m21s
         NEXT_HOP: 10.4.4.4,  Learned from Peer: 10.4.4.4 (1)
         Out-Label: 500000
          LOCAL_PREF: 100,  MED: 2,  ORIGIN: incomplete,  Weight: 0
           AS_PATH:
            Extended Community: RT 300:1 RT 100:2 RT 100:3
2       Prefix: 10.40.1.1/32,  Status: I,  Age: 16h9m21s
        NEXT_HOP: 10.4.4.4,  Learned from Peer: 10.4.4.4 (1)
         Out-Label: 500000
          LOCAL_PREF: 100,  MED: 2,  ORIGIN: incomplete,  Weight: 0
           AS_PATH:
            Extended Community: RT 300:1 RT 100:2 RT 100:3
3       Prefix: 10.40.1.2/32,  Status: I,  Age: 16h9m21s
         NEXT_HOP: 10.4.4.4,  Learned from Peer: 10.4.4.4 (1)
         Out-Label: 500000
          LOCAL_PREF: 100,  MED: 2,  ORIGIN: incomplete,  Weight: 0
           AS_PATH:
            Extended Community: RT 300:1 RT 100:2 RT 100:3
```

TABLE 13 BGP VPNv4/VPNv6 route distinguisher entries

This field...	Displays...
Total number of BGP Routes	The number of routes contained in the BGP4 route table that contain the specified RD.
Prefix	The network address and prefix.
Age	The last time an update occurred.
Learned from Peer	The IP address of the neighbor that sent this route.
Out-Label	MPLS label associated with this device.
MED	The route's metric. When the route does not have a metric, this field is blank.
AS Path	The route's AS path.
Extended Community	Extended community attributes associated with this device.

Displaying VPNv4/VPNv6 neighbor information

To view BGP4 configuration information and statistics for VPNv4/VPNv6 neighbors, enter the following command.

```
          device# show ip bgp vpnv4 neighbors
      Total number of BGP Neighbors: 2
  1   IP Address: 10.2.2.2, AS: 1 (IBGP), RouterID: 10.2.2.2, VRF: default
      State: ESTABLISHED, Time: 14h47m39s, KeepAliveTime: 60, HoldTime: 180
         KeepAliveTimer Expire in 21 seconds, HoldTimer Expire in 141 seconds
         UpdateSource: Loopback 1
         RefreshCapability: Received
      Messages:    Open    Update  KeepAlive Notification Refresh-Req
         Sent    : 1       40      887       0            0
         Received: 1       35      887       0            0
      Last Update Time: NLRI       Withdraw          NLRI       Withdraw
                    Tx: ---        ---           Rx: ---        ---
      Last Connection Reset Reason:Unknown
      Notification Sent:     Unspecified
      Notification Received: Unspecified
      Neighbor NLRI Negotiation:
        Peer Negotiated IPV4  unicast  capability
        Peer Negotiated VPNv4 unicast  capability
        Peer configured for IPV4 unicast  Routes
        Peer configured for VPNv4 unicast  Routes
      TCP Connection state: ESTABLISHED
      TTL check: 0, value: 0, rcvd: 64
         Byte Sent:   29202, Received: 28108
         Local host:  3.3.3.3, Local  Port: 179
         Remote host: 2.2.2.2, Remote Port: 8079
         ISentSeq:    7683960  SendNext:    7713163  TotUnAck:          0
         TotSent:       29203  ReTrans:           0  UnAckSeq:    7713163
         IRcvSeq:   256457831  RcvNext:   256485940  SendWnd:       65000
         TotalRcv:      28109  DupliRcv:          0  RcvWnd:        65000
         SendQue:           0  RcvQue:            0  CngstWnd:       1479
```

This example shows how to display information for VPNv4 neighbors. None of the other display options are used; thus, all of the information is displayed for all neighbors. The number in the far left column indicates the neighbor for which information is displayed. When the user lists information for multiple neighbors, this number makes the display easier to read.

The TCP statistics at the end of the display show status for the TCP session with the neighbor. Most of the fields show information stored in the Transmission Control Block (TCB) for the TCP session between the device and a neighbor. These fields are described in detail in section *3.2 of RFC 793, "Transmission Control Protocol Functional Specification"* .

Syntax: show ip bgp vpnv4 neighbors [*ip-addr* [**advertised-routes** [**detail** [*ip-addr* [**/** *mask-bits*]]]] | [**attribute-entries** [**detail**]] | [**flap-statistics**] | [**last-packet-with-error**] | [**received extended-community**] | [**received prefix-filter**] | [**routes** [**best**] | [**detail** [**best**] | [**not-installed-best**] | [**unreachable**]] | [**rib-out-routes** [*ip-addr/mask-bits* | *ip-addr /net-mask* | **detail**]] | [**routes-summary**]]

The *vrf-name* parameter specifies the VRF whose neighbor the user wants to display information about.

The *ip-addr* option lets the user narrow the scope of the command to a specific neighbor. The display is the same as that for the command without this option except that it is limited to only the neighbor specified.

The **advertised-routes** option displays only the routes that the device has advertised to the neighbor during the current BGP4 neighbor session.

The **attribute-entries** option shows the attribute-entries associated with routes received from the neighbor.

The **flap-statistics** option shows the route flap statistics for routes received from or sent to the neighbor.

The **last-packet-with-error** option displays the last packet from the neighbor that contained an error. The packet's contents are displayed in decoded (human-readable) format.

The **received extended-community** option displays the received extended community Outbound Route Filters (ORFs) received from this neighbor.

The **received prefix-filter** option shows the Outbound Route Filters (ORFs) received from the neighbor. This option applies to cooperative route filtering.

The **routes** option lists the routes received in UPDATE messages from the neighbor. The user can specify the following additional options:

- **best** - Displays the routes received from the neighbor that the device selected as the best routes to their destinations.
- **not-installed-best** - Displays the routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table due to the rib-route-limit (or RTM route table size limit) and always-propagate option to allow the propagating those best BGP routes.
- **unreachable** - Displays the routes that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next hop.
- **detail** - Displays detailed information for the specified routes. The user can refine the information request by also specifying one of the options above (**best** , **not-installed-best** , or **unreachable**).

The **rib-out-routes** option lists the route information base (RIB) for outbound routes. The user can display all the routes or specify a network address.

The **routes-summary** option displays a summary of the following information:

- Number of routes received from the neighbor
- Number of routes accepted by this device from the neighbor
- Number of routes this device filtered out of the UPDATES received from the neighbor and did not accept
- Number of routes advertised to the neighbor
- Number of attribute entries associated with routes received from or advertised to the neighbor

This display shows the following information.

TABLE 14 BGP4 neighbor information

This field...	Displays...
IP Address	The IP address of the neighbor.
AS	The AS the neighbor is in.
EBGP or IBGP	Whether the neighbor session is an IBGP session, an EBGP session, or a confederation EBGP session: • EBGP - The neighbor is in another AS. • EBGP_Confed - The neighbor is a member of another sub-AS in the same confederation. • IBGP - The neighbor is in the same AS.
RouterID	The neighbor's ID.
Description	The description the user gave the neighbor when the user configured it on the device.
State	The state of the session with the neighbor. The states are from the perspective of this device of the session, not the perspective of the neighbor. The state values are based on the BGP4 state machine values described in RFC 1771 and can be one of the following for each device: • IDLE - The BGP4 process is waiting to be started. Usually, enabling BGP4 or establishing a neighbor session starts the BGP4 process. • A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes. • ADMND - The neighbor has been administratively shut down.

TABLE 14 BGP4 neighbor information (continued)

This field...	Displays...
	• A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes. • CONNECT - BGP4 is waiting for the connection process for the TCP neighbor session to be completed. • ACTIVE - BGP4 is waiting for a TCP connection from the neighbor. NOTE When the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection. • OPEN SENT - BGP4 is waiting for an Open message from the neighbor. • OPEN CONFIRM - BGP4 has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. When the device receives a KEEPALIVE message from the neighbor, the state changes to Established. When the message is a NOTIFICATION, the state changes to Idle. • ESTABLISHED - BGP4 is ready to exchange UPDATE messages with the neighbor. • When there is more BGP data in the TCP receiver queue, a plus sign (+) is also displayed. NOTE When the user displays information for the neighbor using the **show ip bgp neighbor** *ip-addr* command, the TCP receiver queue value is greater than 0.
Time	The amount of time this session has been in its current state.
KeepAliveTime	The KeepAliveTime, which specifies how often this device sends keep alive messages to the neighbor.
HoldTime	The hold time, which specifies how many seconds the device waits for a KEEPALIVE or UPDATE message from a BGP4 neighbor before deciding that the neighbor is dead.
PeerGroup	The name of the peer group the neighbor is in, when applicable.
Multihop-EBGP	Whether this option is enabled for the neighbor.
RouteReflectorClient	Whether this option is enabled for the neighbor.
SendCommunity	Whether this option is enabled for the neighbor.
NextHopSelf	Whether this option is enabled for the neighbor.
DefaultOriginate	Whether this option is enabled for the neighbor.
MaximumPrefixLimit	Lists the maximum number of prefixes the device accepts from this neighbor.
RemovePrivateAs	Whether this option is enabled for the neighbor.
RefreshCapability	Whether this device has received confirmation from the neighbor that the neighbor supports the dynamic refresh capability.
CooperativeFilteringCapability	Whether the neighbor is enabled for cooperative route filtering.
Distribute-list	Lists the distribute list parameters, when configured.
Filter-list	Lists the filter list parameters, when configured.
Prefix-list	Lists the prefix list parameters, when configured.
Route-map	Lists the route map parameters, when configured.

TABLE 14 BGP4 neighbor information (continued)

This field...	Displays...
Messages Sent	The number of messages this device has sent to the neighbor. The display shows statistics for the following message types: • Open • Update • KeepAlive • Notification • Refresh-Req
Messages Received	The number of messages this device has received from the neighbor. The message types are the same as for the Message Sent field.
Last Update Time	Lists the last time updates were sent and received for the following: • NLRIs • Withdraws
Last Connection Reset Reason	The reason the previous session with this neighbor ended. The reason can be one of the following: • Reasons described in the BGP specifications: – Message Header Error – Connection Not Synchronized – Bad Message Length – Bad Message Type – OPEN Message Error – Unsupported Version Number – Bad Peer AS Number – Bad BGP Identifier – Unsupported Optional Parameter – Authentication Failure – Unacceptable Hold Time – Unsupported Capability – UPDATE Message Error – Malformed Attribute List – Unrecognized Well-known Attribute – Missing Well-known Attribute – Attribute Flags Error – Attribute Length Error – Invalid ORIGIN Attribute – Invalid NEXT_HOP Attribute – Optional Attribute Error – Invalid Network Field – Malformed AS_PATH – Hold Timer Expired – Finite State Machine Error – Rcv Notification
Last Connection Reset Reason (cont.)	• Reasons specific to the implementation: – Reset All Peer Sessions – User Reset Peer Session – Port State Down – Peer Removed – Peer Shutdown – Peer AS Number Change – Peer AS Confederation Change – TCP Connection KeepAlive Timeout – TCP Connection Closed by Remote – TCP Data Stream Error Detected
Notification Sent	When the device receives a NOTIFICATION message from the neighbor, the message contains an error code corresponding to one of the following

TABLE 14 BGP4 neighbor information (continued)

This field...	Displays...
	errors. Some errors have subcodes that clarify the reason for the error. Where applicable, the subcode messages are listed underneath the error code messages. • Message Header Error: – Connection Not Synchronized – Bad Message Length – Bad Message Type – Unspecified • Open Message Error: – Unsupported Version – Bad Peer As – Bad BGP Identifier – Unsupported Optional Parameter – Authentication Failure – Unacceptable Hold Time – Unspecified • Update Message Error: – Malformed Attribute List – Unrecognized Attribute – Missing Attribute – Attribute Flag Error – Attribute Length Error – Invalid Origin Attribute – Invalid NextHop Attribute – Optional Attribute Error – Invalid Network Field – Malformed AS Path – Unspecified • Hold Timer Expired • Finite State Machine Error • Cease • Unspecified
Notification Received	See above.
TCP Connection state	The state of the connection with the neighbor. The connection can have one of the following states: • LISTEN - Waiting for a connection request. • SYN-SENT - Waiting for a matching connection request after having sent a connection request. • SYN-RECEIVED - Waiting for a confirming connection request acknowledgment after having both received and sent a connection request. • ESTABLISHED - Data can be sent and received over the connection. This is the normal operational state of the connection. • FIN-WAIT-1 - Waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent. • FIN-WAIT-2 - Waiting for a connection termination request from the remote TCP. • CLOSE-WAIT - Waiting for a connection termination request from the local user. • CLOSING - Waiting for a connection termination request acknowledgment from the remote TCP.

TABLE 14 BGP4 neighbor information (continued)

This field...	Displays...
	• LAST-ACK - Waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request). • TIME-WAIT - Waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request. • CLOSED - There is no connection state.
Byte Sent	The number of bytes sent.
Byte Received	The number of bytes received.
Local host	The IP address of the device.
Local port	The TCP port the device is using for the BGP4 TCP session with the neighbor.
Remote host	The IP address of the neighbor.
Remote port	The TCP port the neighbor is using for the BGP4 TCP session with the device.
SentSeq	The initial send sequence number for the session.
SendNext	The next sequence number to be sent.
TotUnAck	The number of sequence numbers sent by the device that have not been acknowledged by the neighbor.
TotSent	The number of sequence numbers sent to the neighbor.
ReTrans	The number of sequence numbers that the device retransmitted because they were not acknowledged.
UnAckSeq	The current acknowledged sequence number.
IRcvSeq	The initial receive sequence number for the session.
RcvNext	The next sequence number expected from the neighbor.
SendWnd	The size of the send window.
TotalRcv	The number of sequence numbers received from the neighbor.
DupliRcv	The number of duplicate sequence numbers received from the neighbor.
RcvWnd	The size of the receive window.
SendQue	The number of sequence numbers in the send queue.
RcvQue	The number of sequence numbers in the receive queue.
CngstWnd	The number of times the window has changed.

Displaying advertised routes for a specified VPNv4 neighbor

To display the routes the device has advertised to a specific VPNv4 neighbor, enter a command such as the following at any level of the CLI.

```
        device# show ip bgp vpnv4 neighbors 10.2.2.2 advertised-routes
     There are 231 routes advertised to neighbor 10.2.2.2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.100.100.30/32   0.0.0.0                    100        0      BE
         AS_PATH: 310
2      10.100.100.29/32   0.0.0.0                    100        0      BE
         AS_PATH: 310
3      10.100.100.28/32   0.0.0.0                    100        0      BE
         AS_PATH: 310
4      10.100.100.27/32   0.0.0.0                    100        0      BE
```

```
          AS_PATH: 310
5       10.100.100.26/32   0.0.0.0                  100        0      BE
          AS_PATH: 310
6       10.100.100.25/32   0.0.0.0                  100        0      BE
          AS_PATH: 310
7       10.100.100.24/32   0.0.0.0                  100        0      BE
          AS_PATH: 310
8       10.100.100.23/32   0.0.0.0                  100        0      BE
          AS_PATH: 310
```

Syntax: show ip bgp vpnv4 neighbor *ip-addr* **advertised-routes** [*ip-addr/prefix*]

Displaying attribute entries for a specified VPNv4/VPNv6 neighbor

The neighbor attribute entries table lists the sets of BGP4 attributes stored in device memory. Each set of attributes is unique and can be associated with one or more routes. In fact, the device typically has fewer route attribute entries than routes. To display the route-attribute entries table for a specified VPNv4 neighbor, enter the following command.

```
          device# show ip bgp vpnv4 neighbors 10.2.2.2 attribute-entries
          Total number of BGP Attribute Entries: 35
1         Next Hop  :0.0.0.0            Metric   :0               Origin:IGP
          Originator:0.0.0.0            Cluster List:None
          Aggregator:AS Number :0      Router-ID:0.0.0.0          Atomic:None
          Local Pref:100               Communities:Internet
          Extended Community: RT 600:1
          AS Path   :310
          Address: 0x247194b0  Hash:45 (0x0100036e) Reference Counts: 0:0:30
2         Next Hop  :0.0.0.0            Metric   :0               Origin:IGP
          Originator:0.0.0.0            Cluster List:None
          Aggregator:AS Number :0      Router-ID:0.0.0.0          Atomic:None
          Local Pref:100               Communities:Internet
          Extended Community: RT 600:1
          AS Path   :311
          Address: 0x2471a480  Hash:47 (0x01000370) Reference Counts: 0:0:30
```

Syntax: show ip bgp vpnv4 neighbors *IPaddress* **attribute-entries**

The *IPaddress* variable is the IP address of the neighbor whose attribute entries the user wants to display.

This display shows the following information.

TABLE 15 BGP4 route-attribute entries information

This field...	Displays...
Total number of BGP Attribute Entries	The number attribute entries in the BGP4 route table for this device.
Next Hop	The IP address of the next hop device for routes with this set of attributes.
Metric	The cost of the routes that have this set of attributes.
Origin	The source of the route information. The origin can be one of the following: • EGP - The routes with this set of attributes came to BGP through EGP. • IGP - The routes with this set of attributes came to BGP through IGP. • INCOMPLETE - The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPF or RIP. When BGP4 compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE.
Originator	The originator of the route in a route reflector environment.

TABLE 15 BGP4 route-attribute entries information (continued)

This field...	Displays...
Cluster List	The route-reflector clusters through which this set of attributes has passed.
Aggregator	Aggregator information: • **AS Number** shows the AS in which the network information in the attribute set was aggregated. This value applies only to aggregated routes and is otherwise 0. • **Router-ID** shows the device that originated this aggregator.
Router ID	
Atomic	Whether the network information in this set of attributes has been aggregated and this aggregation has resulted in information loss: • TRUE - Indicates information loss has occurred • FALSE - Indicates no information loss has occurred **NOTE** Information loss under these circumstances is a normal part of BGP4 and does not indicate an error.
Local Pref	The degree of preference for routes that use this set of attributes relative to other routes in the local AS.
Communities	The communities that routes with this set of attributes are in.
Extended Community	The extended community attributes of the device.
AS Path	The ASs through which routes with this set of attributes have passed. The local AS is shown in parentheses.
Address	This field is for internal Extreme debugging purposes only.
Hash	This field is for internal Extreme debugging purposes only.
Reference Counts	This field is for internal Extreme debugging purposes only.

Displaying received ORFs information for a specified VPNv4/VPNv6 neighbor

To view BGP4 configuration information and statistics for a specified VPNv4 neighbor, enter the following command.

```
      device# show ip bgp vpn neighbors 10.2.2.2 received extended-community Extended-community ORF
capability was not negotiated
 No Prefix filter ORF received from neighbor 10.2.2.2!
```

Displaying a specified neighbor VPNv4/VPNv6 routes

To view the route table for a specified neighbor, enter the following command.

```
        device# show ip bgp vpnv4 neighbors 10.10.2.3 routes
       There are 30 accepted routes from neighbor 10.10.2.3
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.100.100.1/32    10.10.2.3                  100        0      BE
         AS_PATH: 310
2      10.100.100.2/32    10.10.2.3                  100        0      BE
         AS_PATH: 310
3      10.100.100.3/32    10.10.2.3                  100        0      BE
         AS_PATH: 310
4      10.100.100.4/32    10.10.2.3                  100        0      BE
```

```
          AS_PATH: 310
5       10.100.100.5/32    10.10.2.3                100         0       BE
          AS_PATH: 310
6       10.100.100.6/32    10.10.2.3                100         0       BE
          AS_PATH: 310
7       10.100.100.7/32    10.10.2.3                100         0       BE
          AS_PATH: 310
8       10.100.100.8/32    10.10.2.3                100         0       BE
          AS_PATH: 310
9       10.100.100.9/32    10.10.2.3                100         0       BE
          AS_PATH: 310
```

Syntax: show ip bgp vpnv4 neighbors *ip-addr* **routes**

For information about the fields in this display, see the following table.

TABLE 16 BGP4 VPNv4/VPNv6 neighbors information

This field...	Displays...
Total number of BGP VPNv4 Routes:	The number of BGP VPNv4 routes.
Status or Status Codes	The route's status, which can be one or more of the following: • A - AGGREGATE. The route is an aggregate route for multiple networks. • B - BEST. BGP4 has determined that this is the optimal route to the destination. NOTE When the "b" is shown in lowercase, the software was not able to install the route in the IP route table. • b - NOT-INSTALLED-BEST. The routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table due to the rib-route-limit (or RTM route table size limit) and always-propagate option to allow the propagating those best BGP routes. • C - CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation. • D - DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable. • H - HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now. • I - INTERNAL. The route was learned through BGP4. • L - LOCAL. The route originated on this device. • M - MULTIPATH. BGP4 load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B". NOTE When the "m" is shown in lowercase, the software was not able to install the route in the IP route table. • S - SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors. NOTE This field appears only when the user enters the **route** option.

TABLE 16 BGP4 VPNv4/VPNv6 neighbors information (continued)

This field...	Displays...
Origin code	A character the display uses to indicate the route's origin. The origin code appears to the right of the AS path (Path field). The origin codes are described in the command's output.
Route Distinguisher	A unique ID that is prepended on any address being routed or advertised from a VRF. The RD can be defined as either ASN-relative or IP address-relative as described: • ASN-relative - Composed of the local ASN number followed by a ":" and a unique arbitrary number. For example: 3:6 • IP address-relative - Composed of the local IP address followed by a ":" and a unique arbitrary number.
Network	IP address or mask of the destination network of the route.
Next Hop	The next-hop device for reaching the network from this device.
Metric	The value of the route's MED attribute. When the route does not have a metric, this field is blank.
LocPrf	The degree of preference for this route relative to other routes in the local AS. When the BGP4 algorithm compares route on the basis of local preference, the route with the higher local preference is chosen. The preference can have a value from 0-4294967295
Weight	The value that this route associates with routes from a specific neighbor. For example, when the device receives routes to the same destination from two BGP4 neighbors, the device prefers the route from the neighbor with the larger weight.
Path	The AS path of the route.

Displaying the best routes

To display the routes received from a specific neighbor that are the "best" routes to their destinations, enter a command such as the following at any level of the CLI.

```
device# show ip bgp vpnv4 neighbor 192.168.4.211 routes best
```

Syntax: show ip bgp vpnv4 neighbor *ip-addr* **routes best**

Displaying the best routes that were nonetheless not installed in the IP route table

To display the BGP4 routes received from a specific neighbor that are the "best" routes to their destinations but are not installed in the device's IP route table, enter a command such as the following at any level of the CLI.

```
device# show ip bgp vpnv4 neighbor 192.168.4.211 routes not-installed-best
```

Each of the displayed routes is a valid path to its destination, but the device received another path from a different source (such as OSPF, RIP, or a static route) that has a lower administrative distance. The device always selects the path with the lowest administrative distance to install in the IP route table.

Syntax: show ip bgp vpnv4 neighbor *ip-addr* **routes not-installed-best**

Displaying the routes whose destinations are unreachable

To display BGP4 routes whose destinations are unreachable using any of the BGP4 paths in the BGP4 route table, enter a command such as the following at any level of the CLI.

```
device# show ip bgp vpnv4 neighbor 192.168.4.211 routes unreachable
```

Syntax: show ip bgp vpnv4 neighbor *ip-addr* **routes unreachable**

Displaying the Adj-RIB-Out for a VRF neighbor

To display the device's current BGP4 Routing Information Base (Adj-RIB-Out) for a specific VRF neighbor and a specific destination network, enter a command such as the following at any level of the CLI.

```
        device# show ip bgp vpnv4 neighbor 10.10.2.3 rib-out-routes
       There are 154 RIB_out routes for neighbor 10.10.2.3
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.100.101.30/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
2      10.100.101.29/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
3      10.100.101.28/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
4      10.100.101.27/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
5      10.100.101.26/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
6      10.100.101.25/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
7      10.100.101.24/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
8      10.100.101.23/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
9      10.100.101.22/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
10     10.100.101.21/32   10.10.3.3                  100        0      BE
         AS_PATH: 311
```

The Adj-RIB-Out contains the routes that the device either has most recently sent to the VRF neighbor or is about to send to the neighbor.

Syntax: show ip bgp vpnv4 neighbor *ip-addr* **rib-out-routes** [*ip-addr/prefix*]

Displaying routes summary for a specified VPNv4/VPNv6 neighbor

To view the route table for a specified VPNv4 neighbor, enter the following command.

```
      device# show ip bgp vpnv4 neighbor 10.10.2.3 routes-summary
1   IP Address: 10.10.2.3
Routes Accepted/Installed:30,  Filtered/Kept:0,  Filtered:0
   Routes Selected as BEST Routes:30
      BEST Routes not Installed in IP Forwarding Table:0
   Unreachable Routes (no IGP Route for NEXTHOP):0
   History Routes:0
NLRIs Received in Update Message:30,  Withdraws:0 (0),  Replacements:0
   NLRIs Discarded due to
      Maximum Prefix Limit:0, AS Loop:0
      Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
      Duplicated Originator_ID:0, Cluster_ID:0
Routes Advertised:154,  To be Sent:0,  To be Withdrawn:0
NLRIs Sent in Update Message:154,  Withdraws:0,  Replacements:0
Peer Out of Memory Count for:
   Receiving Update Messages:0, Accepting Routes(NLRI):0
   Attributes:0, Outbound Routes(RIB-out):0 Outbound Routes Holder:0
```

This display shows the following information.

TABLE 17 BGP4 route summary information for a VPNv4 neighbor

This field...	Displays...
Routes Accepted or Installed	How many routes the has received from the neighbor during the current BGP4 session: • Filtered - Indicates how many of the received routes the device filtered and did not accept. • Filtered or kept - Indicates how many of the received routes the device did not accept or install because they were denied by filters.
Routes Selected as BEST Routes	The number of routes that the device selected as the best routes to their destinations.
BEST Routes not Installed in IP Forwarding Table	The number of routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the device received better routes from other sources (such as OSPF, RIP, or static IP routes).
Unreachable Routes	The number of routes received from the neighbor that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next hop.
History Routes	The number of routes that are down but are being retained for route flap dampening purposes.
NLRIs Received in Update Message	The number of routes received in Network Layer Reachability (NLRI) format in UPDATE messages: • Withdraws - The number of withdrawn routes the device has received. • Replacements - The number of replacement routes the device has received.
NLRIs Discarded due to	Indicates the number of times the device discarded an NLRI for the neighbor due to the following reasons: • Maximum Prefix Limit - The configured maximum prefix amount had been reached. • AS Loop - An AS loop occurred. An AS loop occurs when the BGP4 AS-path attribute contains the local AS number. • Invalid Nexthop - The next hop value was not acceptable. • Duplicated Originator_ID - The originator ID was the same as the local device ID. • Cluster_ID - The cluster list contained the local cluster ID, or contained the local device ID (see above) when the cluster ID is not configured.
Routes Advertised	The number of routes the device has advertised to this neighbor: • To be Sent - The number of routes the device has queued to send to this neighbor. • To be Withdrawn - The number of NLRIs for withdrawing routes the device has queued up to send to this neighbor in UPDATE messages.
NLRIs Sent in Update Message	The number of NLRIs for new routes the has sent to this neighbor in UPDATE messages: • Withdraws - The number of routes the device has sent to the neighbor to withdraw. • Replacements - The number of routes the device has sent to the neighbor to replace routes the neighbor already has.

TABLE 17 BGP4 route summary information for a VPNv4 neighbor (continued)

This field...	Displays...
Peer Out of Memory Count for	Statistics for the times the device has run out of BGP4 memory for the neighbor during the current BGP4 session: • Receiving Update Messages - The number of times UPDATE messages were discarded because there was no memory for attribute entries. • Accepting Routes (NLRI) - The number of NLRIs discarded because there was no memory for NLRI entries. This count is not included in the Receiving Update Messages count. • Attributes - The number of times there was no memory for BGP4 attribute entries. • Outbound Routes (RIB-out) - The number of times there was no memory to place a "best" route into the neighbor's route information base (Adj-RIB-Out) for routes to be advertised.

Displaying summary route information

To display summary statistics for all the VPNv4 routes in the device's BGP route table, enter a command such as the following at any level of the CLI.

```
      device# show ip bgp vpnv4 routes summary
Total number of BGP routes (NLRIs) Installed     : 184
Distinct BGP destination networks                 : 184
Filtered bgp routes for soft reconfig             : 0
Routes originated by this router                  : 4
Routes selected as BEST routes                    : 184
BEST routes not installed in IP forwarding table : 0
Unreachable routes (no IGP route for NEXTHOP)     : 0
IBGP routes selected as best routes               : 90
EBGP routes selected as best routes               : 90
```

Syntax: show ip bgp vpnv4 routes summary

This display shows the following information.

TABLE 18 BGP VPNv4 summary route information

This field...	Displays...
Total number of BGP VPNv4 routes (NLRIs) Installed	The number of BGP VPNv4 routes the device has installed in the BGP route table.
Distinct BGP VPNv4 destination networks	The number of destination networks the installed routes represent. The BGP route table can have multiple routes to the same network.
Filtered BGP VPNv4 routes for soft reconfig	The number of route updates received from soft-reconfigured neighbors or peer groups that have been filtered out but retained.
Routes originated by this device	The number of VPNv4 routes in the BGP route table that this device originated.
Routes selected as BEST routes	The number of VPNv4 routes in the BGP route table that this device has selected as the best routes to the destinations.
BEST routes not installed in IP forwarding table	The number of BGP VPNv4 routes that are the best BGP VPNv4 routes to their destinations but were not installed in the IP route table because the device received better routes from other sources.
Unreachable routes (no IGP route for NEXTHOP)	The number of routes in the BGP route table whose destinations are unreachable because the next hop is unreachable.
IBGP routes selected as best routes	The number of "best" routes in the BGP VPNv4 route table that are IBGP routes.

TABLE 18 BGP VPNv4 summary route information (continued)

This field...	Displays...
EBGP routes selected as best routes	The number of "best" routes in the BGP VPNv4 route table that are EBGP routes.

Displaying the VPNv4/VPNv6 route table

When the user wants to view all the VPNv4 and VPNv6 routes in a network, the user can display the BGP VPNv4 and VPNv6 table using the following method.

To view the BGP VPNv4 route table, enter the following command.

```
        device# show ip bgp vpnv4 routes
Total number of BGP Routes: 288
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix           Next Hop            Metric     LocPrf     Weight Status
Route Distinguisher: 4:1
1      10.6.1.0/24      10.2.2.2         3          100        0      I
         AS_PATH:
2      10.8.1.0/24      10.2.2.2         2          100        0      I
         AS_PATH:
3      10.40.1.1/32     10.2.2.2         4          100        0      I
         AS_PATH:
4      10.40.1.2/32     10.2.2.2         4          100        0      I
         AS_PATH:
5      10.40.1.3/32     10.2.2.2         4          100        0      I
         AS_PATH:
```

Syntax: show ip bgp vpnv4 routes [*ip-addr*] | *num* | [**age** *secs*] | [**as-path-access-list** *num*] | [**as-path-filter** *num,num,...*] | [**best**] | [**cidr-only**] | [**community** *num* | **no-export** | **no-advertise** | **internet** | **local-as**] | [**community-access-list** *num*] | **community-filter** *num* | **community-reg-expression** *regular-expression* | **detail** | **local** | **neighbor** *ip-addr* [**next-hop** *ip-addr*] | [**no-best**] | [**not-installed-best**] | [**prefix-list** *string*] | [**regular-expression** *regular-expression*] | [**route-map** *map-name*] | [**summary**] | [**unreachable**]

The *ip-addr* option displays routes for a specific network.

The *num* option specifies the table entry with which the user wants the display to start. For example, when the user wants to list entries beginning with table entry 100, specify 100.

The **age***secs* parameter displays only the routes that have been received or updated more recently than the number of seconds the user specifies.

The **as-path-access-list** *num* parameter filters the display using the specified AS-path ACL.

The **best** parameter displays the routes received from the neighbor that the device selected as the best routes to their destinations.

The **cidr-only** option lists only the routes whose network masks do not match their class network length.

The **community** option lets the user display routes for a specific community. The user can specify **local-as** , **no-export** , **no-advertise** , **internet** , or a private community number. The user can specify the community number as either two five-digit integer values of up to 1-65535, separated by a colon (for example, 12345:6789) or a single long integer value.

The **community-access-list** *num* parameter filters the display using the specified community ACL.

The **community-filter** option lets the user display routes that match a specific community filter.

The **community regular-expression** *regular-expression* option filters the display based on a specified community regular expression.

The **local** option....

The **neighbor** *ip-addr* option displays the number of accepted routes from the specified BGP neighbor.

The **detail** option lets the user display more details about the routes. The user can refine the request by also specifying one of the other display options after the **detail** keyword.

The **next-hop** *ip-addr* option displays the routes for a given next-hop IP address.

The **no-best** option displays the routes for which none of the routes to a given prefix were selected as the best route.

The **not-installed-best** option displays the routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table due to the rib-route-limit (or RTM route table size limit) and always-propagate option to allow the propagating those best BGP routes.

The **prefix-list***string* parameter filters the display using the specified IP prefix list.

The **regular-expression** *regular-expression* option filters the display based on a regular expression.

The **route-map***map-name* parameter filters the display using the specified route map. The software displays only the routes that match the match statements in the route map. The software disregards the route map's set statements.

The **summary** option displays summary information for the routes.

The **unreachable** option displays the routes that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next hop.

For information about the fields in this display, see the table below.

TABLE 19 BGP4 VPNv4 information

This field...	Displays...
Total number of BGP VPNv4 Routes:	The number of BGP VPNv4 routes.
Status or Status Codes	The route's status, which can be one or more of the following: • A - AGGREGATE. The route is an aggregate route for multiple networks. • B - BEST. BGP4 has determined that this is the optimal route to the destination. NOTE When the "b" is shown in lowercase, the software was not able to install the route in the IP route table. • b - NOT-INSTALLED-BEST. The routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table due to the rib-route-limit (or RTM route table size limit) and always-propagate option to allow the propagating those best BGP routes. • C - CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation. • D - DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable. • H - HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now. • I - INTERNAL. The route was learned through BGP4. • L - LOCAL. The route originated on this device. • M - MULTIPATH. BGP4 load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".

TABLE 19 BGP4 VPNv4 information (continued)

This field...	Displays...
	NOTE When the "m" is shown in lowercase, the software was not able to install the route in the IP route table. • S - SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors. NOTE This field appears only when the user enters the **route** option.
Origin code	A character the display uses to indicate the route's origin. The origin code appears to the right of the AS path (Path field). The origin codes are described in the command's output.
Route Distinguisher	A unique ID that is prepended on any address being routed or advertised from a VRF. The RD can be defined as either ASN-relative or IP address-relative as described: • ASN-relative - Composed of the local ASN number followed by a ":" and a unique arbitrary number. For example: 3:6 • IP address-relative - Composed of the local IP address followed by a ":" and a unique arbitrary number.
Network	IP address or mask of the destination network of the route.
Next Hop	The next-hop device for reaching the network from this device.
Metric	The value of the route's MED attribute. When the route does not have a metric, this field is blank.
LocPrf	The degree of preference for this route relative to other routes in the local AS. When the BGP4 algorithm compares route on the basis of local preference, the route with the higher local preference is chosen. The preference can have a value from 0-4294967295
Weight	The value that this route associates with routes from a specific neighbor. For example, when the device receives routes to the same destination from two BGP4 neighbors, the device prefers the route from the neighbor with the larger weight.
Path	The AS path of the route.

Displaying the best VPNv4/VPNv6 routes

To display all the VPNv4 routes in the BGP VPNv4 route table for the Extreme device that are the best routes to their destinations, enter a command such as the following at any level of the CLI.

```
        device(config-bgp-router)# show ip bgp vpnv4 routes best
Total number of BGP Routes: 28
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
Route Distinguisher: 4:1
1      3.0.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 80
2      4.0.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 1
3      4.60.212.0/22      192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      6.0.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 3356 7170 1455
5      9.2.0.0/16         192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701
```

Syntax: show ip bgp vpnv4 routes best

For information about the fields in this display, see the Displaying the VPNv4 route table task.

Displaying best VPNv4/VPNv6 routes that are not in the IP route table

When the Extreme device has multiple routes to a destination, the device selects the route with the lowest administrative distance as the best route, and installs that route in the IP route table.

To display the BGP4 routes that are the "best" routes to their destinations but are not installed in the device's IP route table, enter a command such as the following at any level of the CLI.

```
        device(config-bgp-router)# show ip bgp vpnv4 routes not-installed-best
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
Route Distinguisher: 4:1
       Prefix             Next Hop        Metric     LocPrf                    Weight Status
1      10.0.0.0/8                                    192.168.4.106              100        0      BE
       AS_PATH: 65001 4355 701 80
```

Each of the displayed routes is a valid path to its destination, but the device received another path from a different source that has a lower administrative distance. The device always selects the path with the lowest administrative distance to install in the IP route table.

Syntax: show ip bgp vpnv4 routes not-installed-best

For information about the fields in this display, see the Displaying the VPNv4 route table task.

> NOTE
> To display the routes that the Extreme device has selected as the best routes and installed in the IP route table, display the IP route table using the **show ip route** command.

Displaying VPNv4/VPNv6 routes with unreachable destinations

To display BGP VPNv4 routes whose destinations are unreachable using any of the paths in the BGP route table, enter a command such as the following at any level of the CLI.

```
        device(config-bgp-router)# show ip bgp vpnv4 routes unreachable
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
Route Distinguisher: 4:1
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.0.0.0/8                                    192.168.4.106              100            0      BE
       AS_PATH: 65001 4355 701 80
```

Syntax: show ip bgp vpnv4 routes unreachable

For information about the fields in this display, see the Displaying the VPNv4 route table task.

Displaying information for a specific VPNv4/VPNv6 route

To display BGP VPNv4 route information by specifying an IP address within the network, enter a command such as the following at any level of the CLI.

```
        device# show ip bgp vpnv4 routes 10.8.1.0/24
Route Distinguisher: 4:1
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
```

```
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.8.1.0/24        10.2.2.2                              2          100        0      I
         AS_PATH:
Route Distinguisher: 5:1
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop             Metric     LocPrf     Weight Status
1      10.8.1.0/24                                    10.4.4.4         3          100        0      I
         AS_PATH:
```

Syntax: show ip bgp vpnv4 routes *ip-address*/*prefix* [**longer-prefixes** | *ip-addr*]

Displaying VPNv4/VPNv6 route details

Here is an example of the information displayed when the user uses the **detail** option. In this example, the information for one route is shown.

```
      device# show ip bgp vpnv4 routes detail
Total number of BGP Routes: 288
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
Route Distinguisher: 4:1
1      Prefix: 10.6.1.0/24,  Status: I,  Age: 15h36m10s
       NEXT_HOP: 10.2.2.2,  Learned from Peer: 10.2.2.2 (1)
       Out-Label: 500000
         LOCAL_PREF: 100,  MED: 3,  ORIGIN: incomplete,  Weight: 0
         AS_PATH:
         Extended Community: RT 300:1 OSPF DOMAIN ID:0.0.0.0 OSPF RT 0:1:0 OSPF ROUTER ID:0.0.0.0
```

TABLE 20 BGP VPNv4 route information

This field...	Displays...
Prefix	The network address and prefix.
Age	The last time an update occurred.
Learned from Peer	The IP address of the neighbor that sent this route.
Local_Pref	The degree of preference for this route relative to other routes in the local AS. When the BGP4 algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295.
MED	The route's metric. When the route does not have a metric, this field is blank.
Origin	The source of the route information. The origin can be one of the following: • EGP - The routes with this set of attributes came to BGP through EGP. • IGP - The routes with this set of attributes came to BGP through IGP. • INCOMPLETE - The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPF or RIP. When BGP4 compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE.
Atomic	Whether network information in this route has been aggregated and this aggregation has resulted in information loss.

TABLE 20 BGP VPNv4 route information (continued)

This field...	Displays...
	NOTE Information loss under these circumstances is a normal part of BGP4 and does not indicate an error.
Aggregation ID	The router that originated this aggregator.
Aggregation AS	The AS in which the network information was aggregated. This value applies only to aggregated routes and is otherwise 0.
Originator	The originator of the route in a route reflector environment.
Cluster List	The route-reflector clusters through which this route has passed.
Learned From	The IP address of the neighbor from which the Extreme device learned the route.
Admin Distance	The administrative distance of the route.
Adj_RIB_out	The number of neighbors to which the route has been or is advertised. This is the number of times the route has been selected as the best route and placed in the Adj-RIB-Out (outbound queue) for a BGP4 neighbor.
Communities	The communities the route is in.
Extended Community	The device's extended community attributes.

Displaying additional BGP or MPLS VPN information

This section presents a variety of ways to view additional information about a BGP or MPLS configuration on the device.

Displaying VRF information

To display IP Information for a specified VRF, enter the following command at any level of the CLI.

```
        device# show vrf
Total number of VRFs configured: 1
Status Codes - A:active, D:pending deletion, I:inactive
Name                 Default RD                  IFL ID  vrf|v4|v6                    Routes Interfaces
a                         1:1                131071   A | A| A                              14

Total number of IPv4 unicast route for all non-default VRF is 12
Total number of IPv6 unicast route for all non-default VRF is 2

device# show vrf a
VRF a, default RD 1:1, Table ID 1 IFL ID 131071
Label: (Not Allocated),  Label-Switched Mode: OFF
IP Router-Id: 0.0.0.0
  No interfaces
  No Export VPN route-target communities
  No Import VPN route-target communities
  No import route-map
  No export route-map

  Address Family IPv4
    Max Routes: 5120
    Number of Unicast Routes: 12
    No Export VPN route-target communities
    No Import VPN route-target communities
Address Family IPv6
    Max Routes: 128
    Number of Unicast Routes: 2
    No Export VPN route-target communities
    No Import VPN route-target communities
```

Syntax: show vrf *vrf-name*

The *vrf-name* parameter specifies the VRF that the user wants to display IP information for.

TABLE 21 Output from the show VRF command

This field...	Displays...
VRF Name	The name of the VRF.
Default RD	The default route distinguisher for the VRF.
Table ID	The table ID for the VRF.
Routes	The total number of IPv4 and IPv6 Unicast routes configured on this VRF.
Label	Display the unique VRF label that has been assigned to the specified VRF.
Label Switched Mode	Displays when Label Switched Mode is ON or OFF.
Max routes	The maximum number of routes that can be configured on this VRF.
Number of Unicast Routes	The number of Unicast routes configured on this VRF.
Interfaces	The interfaces from this Extreme device that are configured within this VRF.
Export VPN route-target communities:	The export route-targets that are configured for this VRF.
Import VPN route-target communities	The import route-targets that are configured for this VRF.
Import route-map	The name of the import route-map when any that is configured for this VRF.
Export route-map	The name of the export route-map when a route-map has been configured for this VRF.

Displaying the IP route table for a specified VRF

To display the IP routes for a specified VRF, enter the following command at any CLI level.

```
        device# show ip route vrf green
Total number of IP routes: 99
 Type Codes - B:BGP  D:Connected  S:Static  R:RIP  O:OSPF;   Cost - Dist/Metric
        Destination        Gateway         Port        Cost       Type
1       10.5.1.0/24        192.168.201.2   eth 6/3     110/2      O
2       10.6.1.0/24        10.4.4.4                                    lsp toR4    200/0      B
3       10.8.1.0/24        10.2.2.2                                    lsp toR2    200/0      B
4       10.30.1.1/32       192.168.201.2   eth 6/3     110/3      O1
5       10.30.1.2/32       192.168.201.2   eth 6/3     110/3      O1
6       10.30.1.3/32       192.168.201.2   eth 6/3     110/3      O1
7       10.30.1.4/32       192.168.201.2   eth 6/3     110/3      O1
8       10.30.1.5/32       192.168.201.2   eth 6/3     110/3      O1
9       10.30.1.6/32       192.168.201.2   eth 6/3     110/3      O1
10      10.30.1.7/32       192.168.201.2   eth 6/3     110/3      O1
11      10.30.1.8/32       192.168.201.2   eth 6/3     110/3      O1
```

Syntax: show ip route vrf *vrf-name*

The *vrf-name* parameter specifies the VRF that the user wants to display IP routes for.

The following table lists the information displayed by the **show ip route vrf** command.

TABLE 22 CLI display of IP route table

This field...	Displays...
Total number of IP routes	The total number of IP routes that are in the specified VRP routing table.
Destination	The destination network of the route.
NetMask	The network mask of the destination address.

TABLE 22 CLI display of IP route table (continued)

This field...	Displays...
Gateway	The next-hop router.
Port	The port through which this Extreme device sends packets to reach the route's destination.
Cost	The route's cost.
Type	The route type, which can be one of the following: • B - The route was learned from BGP. • D - The destination is directly connected to this Extreme device. • R - The route was learned from RIP. • S - The route is a static route. • * - The route is a candidate default route. • O - The route is an OSPF route. Unless the user uses the **ospf** option to display the route table, "O" is used for all OSPF routes. When the user does use the **ospf** option, the following type codes are used: – O - OSPF intra area route (within the same area). – IA - The route is an OSPF inter area route (a route that passes from one area into another). – E1 - The route is an OSPF external type 1 route. – E2 - The route is an OSPF external type 2 route.

Displaying ARP VRF information

To display the ARP information for a specified VRF, enter the following command.

```
        device# show arp vrf green
Total number of ARP entries: 9
Entries in VRF green:
      IP Address          MAC Address         Type        Age           Port
1     192.168.201.2       2001:DB8.52cf.e840      Dynamic     0         6/3
```

Syntax: show arp vrf *vrf-name* [**number**] [**ip-address**] [**ethernet** *slot/port*] [**mac-address** *mac-addr*]

The *vrf-name* parameter specifies the VRF that the user wants to display arp entries for.

To clear the ARP table.

```
        device# clear arp vrf green
```

Syntax: clear arp vrf *vrf-name*

Displaying OSPF information for a VRF

To display the OSPF Information for a specified VRF, enter the following command at any CLI level.

```
        device# show ip ospf vrf green
OSPF Version Number           Version 2
Router Id                     192.168.201.1
Domain Id                     10.2.2.2
Domain Tag                    10.2.2.2
ASBR Status                   Yes
ABR Status                    Yes         (1)
Redistribute Ext Routes from  BGP
External LSA Counter          96
Originate New LSA Counter     1738
Rx New LSA Counter            173
External LSA Limit            14447047
```

```
Database Overflow Interval    0
Database Overflow State :     NOT OVERFLOWED
RFC 1583 Compatibility :                        Enabled
```

Syntax: show ip ospf vrf *vrf-name* [**area** [*area-id* | **area-ipaddress**]] [**border-routers** *router-id*] [**config**] [**database** [**database-summary** | **external-link-state** [**advertise** *number*] | **extensive** | **link-state-id** *id-number* | **router-id** *advertising-router-id* | **sequence-number** *HEX*] [**link-state** [**advertise** *number*] | **sabre**

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF information for.

Displaying OSPF area information for a VRF

To display OSPF Area Information for a specified VRF, enter the following command at any level of the CLI.

```
        device# show ip ospf vrf green area
Indx Area            Type   Cost       SPFR       ABR   ASBR  LSA   Chksum(Hex)
1    0               normal 0          6          0     0     6     00039ba2
2    1               normal 0          6          0     2     6     0003af4b
```

Syntax: show ip ospf vrf *vrf-name* **area** [*area-id*] | [*ip-address*]

The *vrf-name* parameter specifies the VRF that the user wants to the OSPF area information for.

The *area-id* parameter shows information for the specified area.

The *ip-address* parameter displays the entry that corresponds to the IP address the user enters.

Displaying OSPF ABR and ASBR information for a VRF

To display OSPF ABR and ABSR Information for a specified VRF, enter the following command at any level of the CLI.

```
        device# show ip ospf vrf green border-routers
      router ID       router type next hop router outgoing interface Area
1     10.2.10.2                          ASBR        192.168.201.2   6/3               1
1     10.5.1.3        ASBR                                   192.168.201.2   6/3               1
```

Syntax: show ip ospf vrf *vrf-name* **border-routers** *router-id*

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF ABR and ABSR information for.

The *router-id* parameter specifies the display of OSPF ABR and ABSR information for the router with the specified router ID.

Displaying general OSPF configuration information for a VRF

To display OSPF ABR and ABSR Information for a specified VRF, enter the following command at any level of the CLI.

```
        device# show ip ospf vrf green config
Router OSPF: Enabled
Redistribution: Enabled
Default OSPF Metric: 10
OSPF Auto-cost Reference Bandwidth: Disabled

OSPF Redistribution Metric: Type2

OSPF External LSA Limit: 14447047

OSPF Database Overflow Interval: 0

RFC 1583 Compatibility: Enabled

Router id: 192.168.201.1
Interface State Change Trap:                    Enabled
```

```
Virtual Interface State Change Trap:              Enabled
Neighbor State Change Trap:                       Enabled
Virtual Neighbor State Change Trap:               Enabled
Interface Configuration Error Trap:               Enabled
Virtual Interface Configuration Error Trap:       Enabled
Interface Authentication Failure Trap:            Enabled
Virtual Interface Authentication Failure Trap:    Enabled
Interface Receive Bad Packet Trap:                Enabled
Virtual Interface Receive Bad Packet Trap:        Enabled
Interface Retransmit Packet Trap:                 Disabled
Virtual Interface Retransmit Packet Trap:         Disabled
Originate LSA Trap:                               Disabled
Originate MaxAge LSA Trap:                        Disabled
Link State Database Overflow Trap:                Disabled
Link State Database Approaching Overflow Trap:    Disabled

OSPF Area currently defined:
Area-ID          Area-Type Cost
0                normal    0
1                normal    0
```

Syntax: show ip ospf vrf *vrf-name* **config**

The *vrf-name* parameter specifies the VRF that the user wants to display general OSPF configuration information for.

Displaying OSPF external link state information for a VRF

To display OSPF External Link State Information for a specified VRF, enter the following command at any level of the CLI.

```
        device# show ip ospf vrf green database external-link-state
Index Aging  LS ID           Router          Netmask  Metric    Flag
1     491    10.30.1.6       10.5.1.3        ffffffff 00000001  0000
2     1005   10.40.1.30      192.168.201.1   ffffffff 8000000a  0000
3     765    10.60.1.10      192.168.201.1   ffffffff 8000000a  0000
4     1005   10.40.1.9       192.168.201.1   ffffffff 8000000a  0000
5     491    10.30.1.19      10.5.1.3        ffffffff 00000001  0000
6     765    10.60.1.23      192.168.201.1   ffffffff 8000000a  0000
7     1005   10.40.1.22      192.168.201.1   ffffffff 8000000a  0000
8     765    10.60.1.2       192.168.201.1   ffffffff 8000000a  0000
9     1005   10.40.1.1       192.168.201.1   ffffffff 8000000a  0000
10    491    10.30.1.11      10.5.1.3        ffffffff 00000001  0000
11    765    10.60.1.15      192.168.201.1   ffffffff 8000000a  0000
12    1005   10.40.1.14      192.168.201.1   ffffffff 8000000a  0000
13    491    10.30.1.24      10.5.1.3        ffffffff 00000001  0000
14    491    10.30.1.3       10.5.1.3        ffffffff 00000001  0000
```

Syntax: show ip ospf vrf *vrf-name* **database external-link-state** [**advertise** *num*] | [**extensive**] | [**link-state-id** *ip-addr*] | [**router-id** *ip-addr*] | [**sequence-number** *num(Hex)*] | [**status** *num*]

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF external link state information for.

The **advertise***num* parameter displays the data in the specified LSA packet. The *num* parameter identifies the LSA packet by its position in the Extreme device's External LSA table. To determine an LSA packet's position in the table, enter the **show ip ospf vrf** *vrf-name***external-link-state** command to display the table.

The **extensive** option displays the data in the LSAs in decrypted format.

The **link-state-id***ip-addr* parameter displays the External LSAs for the LSA source specified by *IP-addr* .

The **router-id***ip-addr* parameter shows the External LSAs for the specified OSPF router.

The **status***num* option shows status information.

The **sequence-number***num (Hex)* parameter displays the External LSA entries for the specified hexadecimal LSA sequence number.

Displaying OSPF link state information for a VRF

To display OSPF Link State Information for a specified VRF, enter the following command at any level of the CLI.

```
        device# show ip ospf vrf green database link-state
Index Area ID         Type LS ID           Adv Rtr         Seq(Hex) Age  Cksum
1     0               Summ 10.2.10.2                              192.168.201.1   8000001b 1145 0x03fb
2     0               Summ 192.168.201.0   192.168.201.1   8000001b 1145 0x4d8d
3     0               Summ 10.8.1.0        192.168.201.1   8000001b 905  0xadc5
4     0               Summ 10.5.1.0        192.168.201.1   8000001b 1145 0xea12
5     0               ASBR 10.2.10.2                              192.168.201.1   8000001b 1145 0xf409
6     0               ASBR 10.5.1.3        192.168.201.1   8000001b 1145 0xbe3a
7     1               Rtr  192.168.201.1   192.168.201.1   80000088 1145 0xf304
8     1               Rtr  10.2.10.2                           10.2.10.2
800000eb 581  0x503d
9     1               Rtr  10.5.1.3        10.5.1.3        8000005e 1470 0xf8b0
10    1               Net  192.168.201.1   192.168.201.1   8000001f 1145 0xb5da
11    1               Net  10.5.1.1        10.2.10.2                       8000004e 1792 0x0fbb
12    1               Summ 10.8.1.0        192.168.201.1   8000001b 905  0xadc5
```

Syntax: show ip ospf vrf *vrf-name* **database link-state** [**advertise** *num*] | [**asbr**] | [**extensive**] | [**link-state-id** *ip-addr*] | [**network**] | [**nssa**] | [**opaque-area**] | [**router**] | [**router-id** *ip-addr*] | [**sequence-number** *num(Hex)*] | [**status** *num*] | [**summary**]

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF link state information for.

The **advertise** *num* parameter displays the hexadecimal data in the specified LSA packet. The *num* parameter identifies the LSA packet by its position in the Extreme device's External LSA table. To determine an LSA packet's position in the table, enter the **show ip ospf vrf** *vrf-name***external-link-state** command to display the table.

The **asbr** option shows ASBR information.

The **extensive** option displays the LSAs in decrypted format.

The **link-state-id***ip-addr* parameter displays the External LSAs for the LSA source specified by *IP-addr* .

The **network** option shows network information.

The **nssa** option shows network information.

The **opaque-area** option shows information for opaque areas.

The **router-id***ip-addr* parameter shows the External LSAs for the specified OSPF router.

The **sequence-number***num (Hex)* parameter displays the External LSA entries for the specified hexadecimal LSA sequence number.

The **status***num* option shows status information.

The **summary** option shows summary information.

Displaying OSPF interface information

To display OSPF interface information for a specified VRF, enter the following command at any CLI level.

```
        device# show ip ospf vrf green interface
ethernet 6/3,OSPF enabled
     IP Address 192.168.201.1, Area 1
     OSPF state DR, Pri 1, Cost 1, Options 2, Type broadcast Events 3
     Timers(sec): Transit 1, Retrans 5, Hello 10, Dead 40
     DR:  Router ID 192.168.201.1     Interface Address 192.168.201.1
     BDR: Router ID 1.2.10.2          Interface Address 192.168.201.2
     Neighbor Count = 1, Adjacent Neighbor Count= 1
     Neighbor:         192.168.201.2
     Authentication-Key:None
     MD5 Authentication: Key None, Key-Id None, Auth-change-wait-time 300
```

Syntax: show ip ospf vrf *vrf-name* **interface** [*ip-addr*]

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF interface information for.

The *ip-addr* parameter displays the OSPF interface information for the specified IP address.

Displaying OSPF neighbor information for a VRF

To display OSPF neighbor information for a specified VRF, enter the following command at any CLI level.

```
          device# show ip ospf vrf green neighbor

Port  Address         Pri State      Neigh Address   Neigh ID        Ev Opt Cnt
6/3   192.168.201.1   1   FULL/BDR   192.168.201.2   10.2.10.2                              6  2   0
```

Syntax: show ip ospf vrf *vrf-name* **neighbor** [**router-id** *ip-addr*] | [*num*] | [**detail**]

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF neighbor information for.

The **router-id***ip-addr* parameter displays only the neighbor entries for the specified router.

The *num* parameter displays only the entry in the specified index position in the neighbor table. For example, when the user enters "1", only the first entry in the table is displayed.

The **detail** parameter displays detailed information about the neighbor routers.

Displaying the routes that have been redistributed into OSPF

The user can display the routes that have been redistributed into OSPF for a VRF. To display the redistributed routes, enter the following command at any level of the CLI.

```
        device# show ip ospf vrf green redistribute route
  10.6.1.0 10.255.255.0 bgp
  10.8.1.0 10.255.255.0 bgp
  10.40.1.1 10.255.255.255 bgp
  10.40.1.2 10.255.255.255 bgp
```

In this example, four routes have been redistributed from BGP routes.

Syntax: show ip ospf vrf *vrf-name* **redistribute route**

The *vrf-name* parameter specifies the VRF that the user wants to display routes redistributed into OSPF for.

Displaying OSPF route information for a VRF

To display the OSPF route information for a specified VRF, enter the following command at any level of the CLI.

```
          device# show ip ospf vrf green routes
OSPF Area 0x00000001 ASBR Routes 2:
      Destination     Mask            Path_Cost Type2_Cost Path_Type
      10.2.10.2                             10.255.255.255      1         0          Intra
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      10.2.10.2                             10.2.10.2                        Asbr      Valid
0                                   0000
      Paths Out_Port  Next_Hop        Type      State
      1     6/3       192.168.201.2   OSPF      00 00
```

In this example, four routes have been redistributed from BGP routes.

Syntax: show ip ospf vrf *vrf-name* **routes** [*ip-addr*]

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF routes for.

The *ip-addr* parameter specifies a destination IP address. When the user uses this parameter, only the route entries for that destination are shown.

Displaying OSPF sham links

To display the OSPF sham links information for a VRF, enter the **show ip ospf vrf** *vrf-name***sham-links** command at any level of the CLI, as in the following example.

```
        device# show ip ospf vrf CustomerA sham-links
Sham Link in OSPF instance CustomerA to 10.1.1.2 is UP, Established over lsp(LDP)
 Area 1 source address 10.1.1.1
   Link cost 1 Transmit Delay is 1 sec, State ptpt
   Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
   Adjacency State UP, number of interface events 417
```

Syntax: show ip ospf vrf *vrf-name* **sham-links**

The *vrf-name* variable identifies the VRF for which the user wants to display OSPF sham links information.

Displaying OSPF trap status for a VRF

To display the state (enabled or disabled) of the OSPF traps for a specified VRF, enter the following command at any CLI level.

```
        device# show ip ospf vrf green trap
Interface State Change Trap:                    Enabled
Virtual Interface State Change Trap:            Enabled
Neighbor State Change Trap:                     Enabled
Virtual Neighbor State Change Trap:             Enabled
Interface Configuration Error Trap:             Enabled
Virtual Interface Configuration Error Trap:     Enabled
Interface Authentication Failure Trap:          Enabled
Virtual Interface Authentication Failure Trap:  Enabled
Interface Receive Bad Packet Trap:              Enabled
Virtual Interface Receive Bad Packet Trap:      Enabled
Interface Retransmit Packet Trap:               Disabled
Virtual Interface Retransmit Packet Trap:       Disabled
Originate LSA Trap:                             Disabled
Originate MaxAge LSA Trap:                      Disabled
Link State Database Overflow Trap:              Disabled
Link State Database Approaching Overflow Trap:  Disabled
```

Syntax: show ip ospf vrf *vrf-name* **trap**

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF trap status for.

Displaying OSPF virtual links for a VRF

To display the OSPF virtual links information for a specified VRF, enter the following command at any level of the CLI.

```
        device# show ip ospf vrf green virtual-link
No ospf virtual-link entries available
```

Syntax: show ip ospf vrf *vrf-name* **virtual-link** [*num*]

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF virtual links information for.

The *num* parameter displays the table beginning at the specified entry number.

Displaying OSPF virtual neighbor information for a VRF

To display the OSPF virtual neighbor information for a specified VRF, enter the following command at any level of the CLI.

```
device# show ip ospf vrf green virtual neighbor
```

Syntax: show ip ospf vrf *vrf-name* **virtual neighbor** [*num*]

The *vrf-name* parameter specifies the VRF that the user wants to display OSPF virtual neighbor information for.

The *num* parameter displays the table beginning at the specified entry number.

Displaying IP extcommunity list information

To display the IP Extcommunity information, enter the following command at any level of the CLI.

```
device# show ip extcommunity-list
ip extcommunity access list 20:
  permit RT 100:1
```

Syntax: show ip extcommunity-list

For information about the fields, refer to the following.

TABLE 23 Output of show IP extcommunity list

This field...	Displays...
ip extcommunity access list	The contents of all extended community lists on the Extreme device.

Displaying the IP static route table for a VRF

To display the IP static route table for a VRF, enter the following command at any level of the CLI.

```
device# show ip static route vrf green
IP Static Routing Table - entries:
    IP Prefix                    Next Hop                Interface                 Dis/Met/
Tag                Name
    10.22.66.0/24                    10.22.66.0                -                1/1/0
green
```

Syntax: show ip static route vrf *vrf-name*

The *vrf-name* parameter specifies the VRF that the user wants to display the static route table for.

Show run displays the entire name of the static IP route. The **show ip static route** command displays an asterisk (*) after the first twelve characters when the assigned name is thirteen characters or more. The **show ipv6 static route** command displays an asterisk after the first two characters when the assigned name is three characters or more.

Displaying the static ARP table for a VRF

To display the static ARP table for a VRF, enter the following command at any level of the CLI.

```
device# show ip static-arp vrf green
Static ARP table size: 2048, configurable from 2048 to 4096
  Index              IP Address          MAC Address          Port
    1              10.95.6.111                  2001:DB8.093b.d210      1/1
    3              10.95.6.123                2001:DB8.093b.d211      1/1
```

Syntax: show ip static-arp vrf *vrf-name*

The *vrf-name* parameter specifies the VRF that the user wants to display the static ARP table for.

To clear the static ARP table in a VRF, enter the following command.

```
device# clear arp vrf blue
```

Syntax: clear arp vrf *vrf-name*

Displaying TCP connections for a VRF

The **show ip tcp vrf connections** command displays information about each TCP connection on the VRF, including the local IP address, local port number, remote IP address, remote port number and the state of the connection. For example.

```
        device# show ip tcp vrf green connections
Local IP address:port <-> Remote IP address:port TCP state    (hdl itc cln pdn)
0.0.0.0:179 <-> 0.0.0.0:0 LISTEN (000100bf: 13, 0, 0)
Total 1 TCP connections
```

Syntax: show ip tcp vrf *vrf-name* **connections**

The *vrf-name* parameter specifies the VRF that the user wants to display TCP connections for.

Displaying IP route information for a VRF

Display IP route information for a specified VRF by entering the following command.

> NOTE
> When BGP and static routes use an MPLS tunnel as the outgoing interface, the Gateway field displays DIRECT in the output of the **show ip route vrf** *vrf-name* command. This is only applicable when displaying IPv4 routes.

```
        device# show ip route vrf yellow
Total number of IP routes: 2
 Type Codes - B:BGP  D:Connected  S:Static  R:RIP  O:OSPF;   Cost - Dist/Metric
        Destination        Gateway         Port        Cost       Type
1       10.8.8.8/32                                    DIRECT          loopback 1  0/0        D
2       10.9.9.8/32                                    DIRECT          lsp to1     200/0      B
```

Syntax: show ip route vrf *vrf-name* [*num* | *ip-addr* | **bgp** | **connected** | **isis** | **ospf** | **rip** | **static** | **tags**]

The *vrf-name* parameter specifies the VRF that the user wants to display IP route information for.

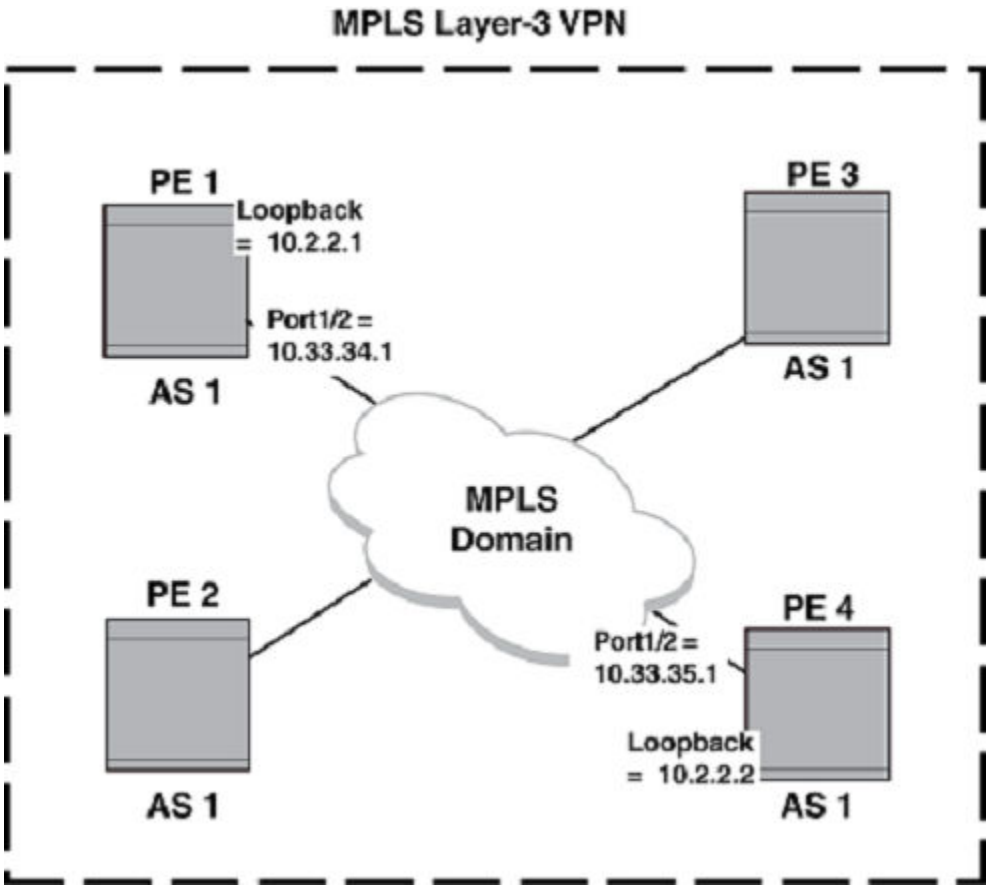
BGP or MPLS VPN sample configurations

This section presents examples of typical MPLS configurations.

Basic configuration example for IBGP on the PEs

PE routers use IBGP to exchange VRF routes. As in all BGP configurations, this is accomplished by configuring BGP neighbors where the user wants to exchange routes. When the neighbors are configured in the same AS, it is an IBGP configuration. In addition, because MPLS LSPs are made between router loopback addresses, the [update-source loopback] parameters must be used. The following diagram shows two PE routers (PE 1 and PE 4) that are configured as BGP neighbors.

FIGURE 36 IBGP example



To configure IBGP on a Provider Edge router (PE) of a BGP or MPLS VPN network, the user must perform the configuration steps listed below.

1. Assigning an AS number to a PE on page 290
2. Assigning a loopback interface on page 291
3. Configuring an IBGP neighbor on a PE on page 291

Assigning an AS number to a PE

In the IBGP configuration used in a BGP or MPLS VPN, all PEs are configured with the same AS number. To assign the local AS number 1 to the PE 1 router as shown in Figure 36 on page 290, enter the following commands.

```
device(config)# router bgp
device(config-bgp)# local-as 1
```

Assigning a loopback interface

A loopback interface is used as the termination for address for BGP sessions. This allows BGP to stay up even when the outbound interface is down as long as an alternate path is available. To install the loopback interface on PE 1 as shown in Figure 36 on page 290, enter the following commands.

```
      device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.2.2.1/32
```

Configuring an IBGP neighbor on a PE

Other PEs that the user wants to exchange IBGP routes with must be configured as BGP neighbors. In addition, the neighbor must be set to enable the BGP to update the loopback address. To assign an IBGP neighbor with the IP address 10.33.35.1, a remote AS number of 1, and an update-source to loopback 1 of the PE 1 router shown in Figure 36 on page 290, enter the following commands.

```
      device(config-bgp)# neighbor 10.2.2.2 remote-as 1
device(config-bgp)# neighbor 10.2.2.2 update-source loopback 1
device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.2.2.2 activate
```

Configuring EBGP on a CE router

Allows route exchanges between a CE router and its associated PE router by enabling BGP on a customer edge (CE) router and configuring an associated premises edge (PE) router as a BGP neighbor.

The following task shows the steps required for enabling BGP on a CE device and assigning a PE device as a BGP neighbor. For an example of a full configuration required to exchange routes in an external BGP (EBGP) network, see the EBGP for route exchange task.

1. On a CE device, enter the **configure terminal** command.

   ```
   device# configure terminal
   ```

2. Enter the **router bgp** command to enable BGP routing.

   ```
   device(config)# router bgp
   ```

3. Enter the **local-as** command to configure the AS number.

   ```
   device(config-bgp)# local-as 2
   ```

4. Configure a PE BGP neighbor using the **neighbor remote-as** command.

   ```
   device(config-bgp)# neighbor 10.33.33.3 remote-as 1
   ```

The following example enables BGP on a CE and assigns a PE as a BGP neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp)# local-as 2
device(config-bgp)# neighbor 10.33.33.3 remote-as 1
```

Configuring EBGP on a PE router

Allows route exchange between a VRF on a PE router and its associated customer edge (CE) router by enabling BGP on the appropriate VRF of the premises edge (PE) router and configuring the associated CE router as a BGP neighbor.

In this task, a CE is assigned as a BGP neighbor to the VRF VPN1 on a PE device. For an example of a full configuration required to exchange routes in an external BGP (EBGP) network, see the EBGP for route exchange task.

1. On a PE device, enter the **configure terminal** command.

```
device# configure terminal
```

2. Enter the **router bgp** command to enable BGP routing.

```
device(config)# router bgp
```

3. Enter the **address-family ipv4 unicast** command to assign a VRF.

```
device(config-bgp)# address-family ipv4 unicast vrf VPN1
```

4. Enter the **neighbor remote-as** command to configure a BGP neighbor (a CE device) to the VRF.

```
device(config-bgp-ipv4u-vrf)# neighbor 10.33.33.2 remote-as 2
```

The following example assigns a CE device as a BGP neighbor to the VRF VPN1 on a PE device.

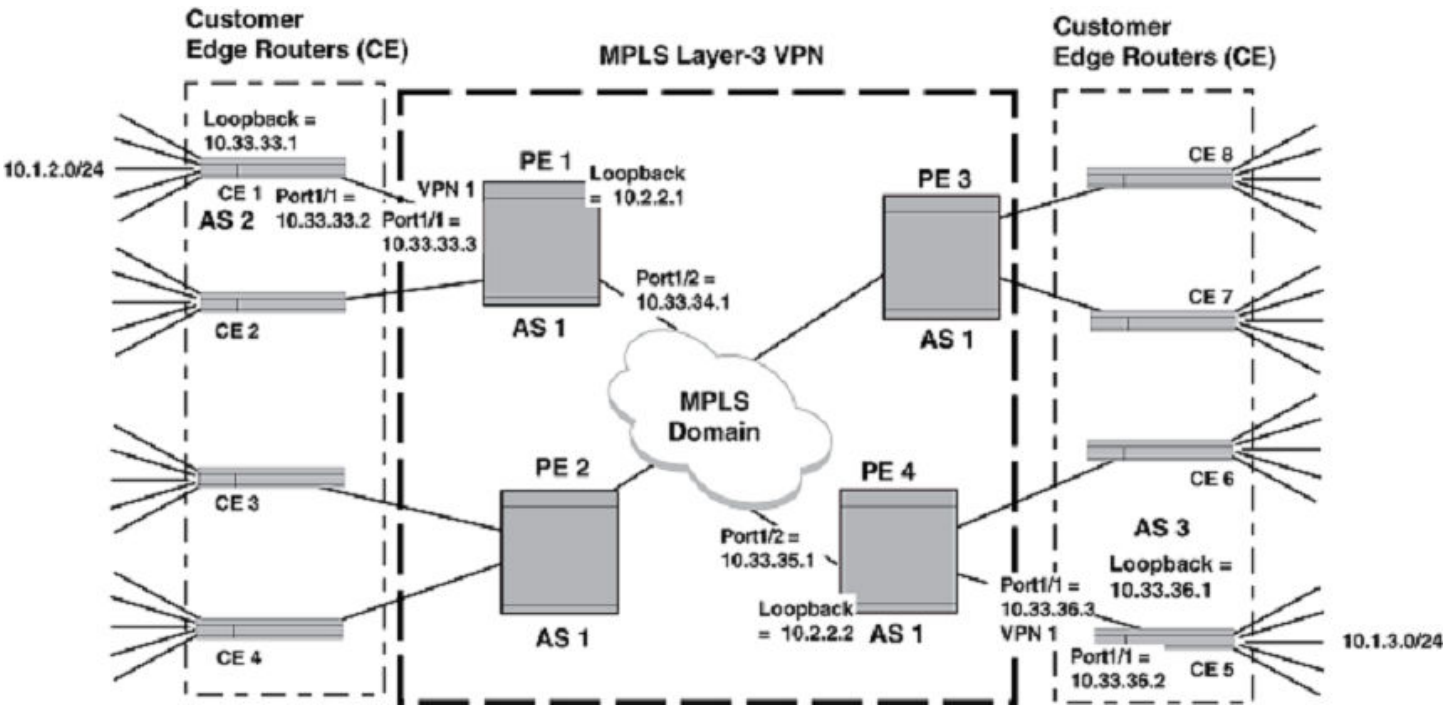
```
device# configure terminal
device(config)# router bgp
device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# neighbor 10.33.33.2 remote-as 2
```

EBGP for route exchange

External BGP (EBGP) can be used to exchange routes from CE routers to PE routers.

To exchange routes, a BGP neighbor must be configured on both CE and PE routers. In the diagram shown below, the CE 1 router is configured to exchange routes with the PE 1 router and the CE 5 router is configured to exchange routes with the PE 4 router.

FIGURE 37 EBGP to CE network example



EBGP to CE network example

In the example shown in the diagram above, the network is configured to use EBGP to forward routes between the networks attached to the CE routers and the PE routers. IBGP is used to forward routes between the PE routers and an LSP tunnel is configured across the MPLS domain. The diagram above contains all of the network addresses and AS numbers required to perform this configuration. The configurations are shown for only the CE 1, CE 5, PE 1 and PE 5 routers, which demonstrates what is required to make VPN1 work. The configurations for VPN2, VPN3, and VPN 4 would essentially be the same.

CE 1 configuration

This configuration example describes what is required to operate the CE 1 router in the diagram above. In this example, a static route is configured between the external network (10.1.2.0/24) and the loopback interface. EBGP is configured between CE 1 and PE 1, and the static route is redistributed through this connection.

```
device(config)# ip route 10.1.2.0/24 10.33.33.1
device(config)# router bgp
device(config-bgp)# local-as 2
device(config-bgp)# neighbor 10.33.33.3 remote-as 1
device(config-bgp)# redistribute static
device(config-bgp)# exit
device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# ip address 10.33.33.2/24
device(config-if-e10000-1/1)# exit
```

CE 5 configuration

This configuration example describes what is required to operate the CE 5 router in the diagram above. In this example, a static route is configured between the external network (10.1.3.0/24) and the loopback interface. EBGP is configured between CE 5 and PE 4, and the static route is redistributed through this connection.

```
device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.33.36.1/32
device(config-lbif-1)# exit
device(config)# ip route 10.1.3.0/24 10.33.36.1
device(config)# router bgp
device(config)# local-as 3
device(config-bgp)# neighbor 10.33.36.3 remote-as 1
device(config-bgp)# redistribute static
device(config-bgp)# exit
device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# ip address 10.33.36.2/24
device(config-if-e10000-1/1)# exit
```

PE 1 configuration

This configuration example describes what is required to operate the PE 1 router in the diagram above. In this example, the VRF VPN1 is created with a unique route descriptor consisting of the BGP AS number (1) and a random other number (2), and route targets are set for import and export. The VRF (VPN1) is defined on the interface that connects to CE 1. EBGP is configured between VPN1 and CE 1. IBGP with extended community attributes is configured between PE 1 and PE 4.

The OSPF area is specified as 0 and MPLS is set up for OSPF traffic engineering. In addition, MPLS is configured with a signaled LSP named "tunnel1" to PE 4.

```
device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.2.2.1/32
device(config-lbif-1)# ip ospf area 0
device(config-lbif-1)# exit

device(config)# vrf VPN1
device(config-vrf-vpn1)# rd 1:1
device(config-vrf-vpn1)# route-target export 100:1
device(config-vrf-vpn1)# route-target import 100:2
device(config-vrf-vpn1)# exit-vrf

device(config)# router bgp
device(config-bgp)# local-as 1
device(config-bgp)# neighbor 10.2.2.2 remote-as 1
device(config-bgp)# neighbor 10.2.2.2 update-source loopback 1
device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.2.2.2 activate
device(config-bgp-vpnv4u)# exit

device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# neighbor 10.33.33.2 remote-as 2
device(config-bgp-ipv4u-vrf)# exit

device(config)# router ospf
device(config-ospf-router)# area 0
device(config-ospf-router)# exit

device(config)# interface ethernet 1/2
device(config-if-e10000-1/2)# ip address 10.33.34.1/24
device(config-if-e10000-1/2)# ip ospf area 0
device(config-if-e10000-1/2)# exit

device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# traffic-engineering ospf
device(config-mpls)# mpls-interface eth 1/2
device(config-mpls)# lsp tunnel1
device(config-mpls-lsp-tunnel1)# to 10.2.2.2
device(config-mpls-lsp-tunnel1)# enable
device(config-mpls-lsp-tunnel1)# exit

device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# vrf forwarding VPN1
device(config-if-e10000-1/1)# ip address 10.33.33.3/24
```

PE 4 configuration

This configuration example describes what is required to operate the PE 2 router in the diagram above. In this example, the VRF VPN1 is created with a unique route descriptor consisting of the BGP AS number (1) and a random other number (2), and route targets are set for import and export. The VRF (VPN1) is defined on the interface that connects to CE 5. EBGP is configured between VPN1 and CE 5. IBGP with extended community attributes is configured between PE 4 and PE 1.

The OSPF area is specified as 0 and MPLS is set up for OSPF traffic engineering. In addition, MPLS is configured with a signaled LSP named "tunnel1" to PE 1.

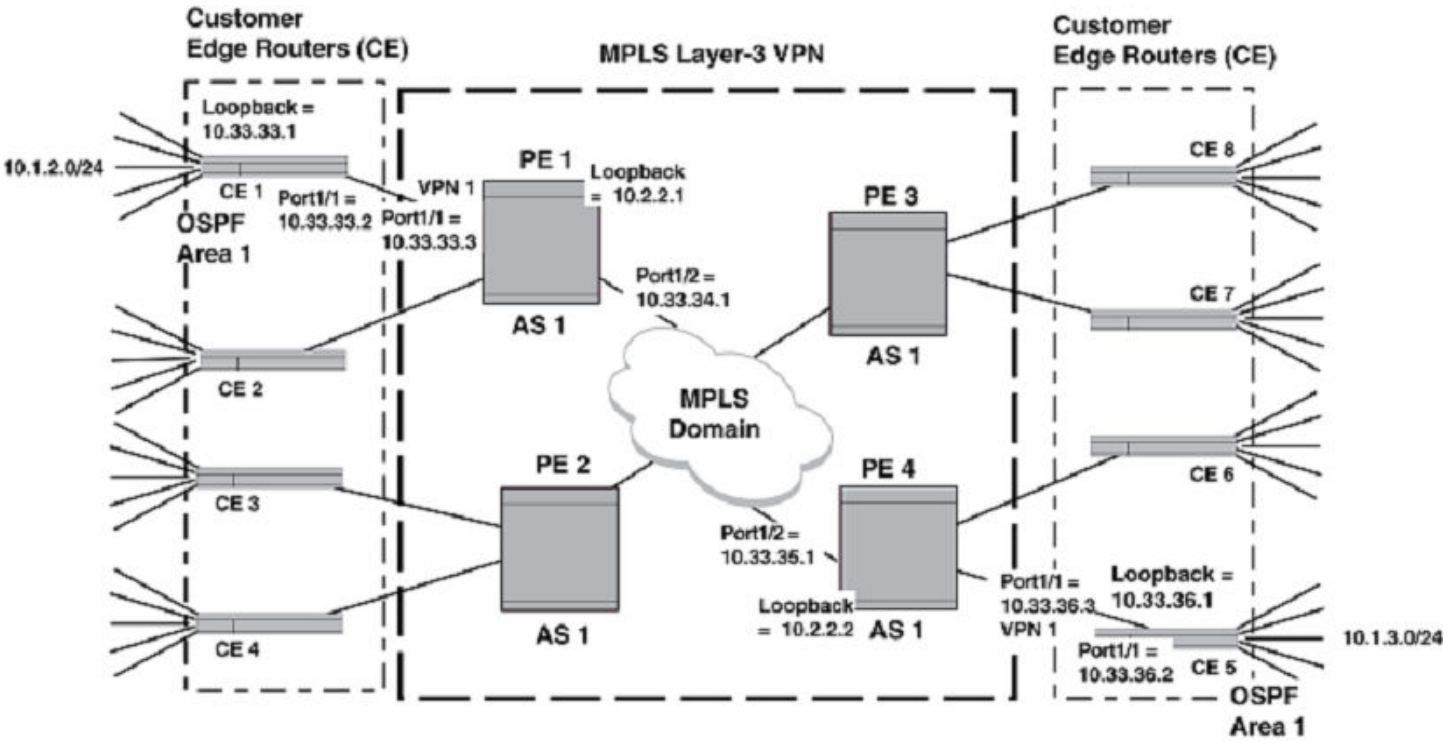
```
device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.2.2.2/32
device(config-lbif-1)# ip ospf area 0
device(config-lbif-1)# exit

device(config)# vrf VPN1
device(config-vrf-vpn1)# rd 1:2
device(config-vrf-vpn1)# route-target export 100:2
device(config-vrf-vpn1)# route-target import 100:1
device(config-vrf-vpn1)# exit-vrf

device(config)# router bgp
device(config-bgp)# local-as 1
device(config-bgp)# neighbor 10.2.2.1 remote-as 1
device(config-bgp)# neighbor 10.2.2.1 update-source loopback 1
device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.2.2.1 activate
device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# neighbor 10.33.36.2 remote-as 2
device(config-bgp-ipv4u-vrf)# exit

device(config)# router ospf
device(config-ospf-router)# area 0
device(config-ospf-router)# exit

device(config)# interface ethernet 1/2
device(config-if-e10000-1/2)# ip address 10.33.35.1/24
device(config-if-e10000-1/2)# ip ospf area 0
device(config-if-e10000-1/2)# exit

device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# traffic-engineering ospf
device(config-mpls)# mpls-interface eth 1/2
device(config-mpls)# lsp tunnel1
device(config-mpls-lsp-tunnel1)# to 10.2.2.1
device(config-mpls-lsp-tunnel1)# enable
device(config-mpls-lsp-tunnel1)# exit

device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# vrf forwarding VPN1
device(config-if-e10000-1/1)# ip address 10.33.36.3/24
device(config-if-e10000-1/1)# exit
```

Static routes for route exchange

Static routes can be used to exchange routes between CE routers and PE routers.

To exchange routes, a default static route must be configured on a CE router to its associated PE router. A static route must also be configured between the PE router and the network (or networks) that the PE wants to advertise as available through a VRF. In this task, the network shown below is configured for a default static route to forward routes between the networks attached to the CE routers and the PE routers. IBGP is used to forward routes between the PE routers and an LSP tunnel is configured across the MPLS domain. The diagram below contains all of the network addresses and AS numbers required to perform this configuration. The configurations are shown for only the CE 1, CE 5, PE 1 and PE 5 routers which demonstrates what is required to make VPN1 work. The configurations for VPN2, VPN3, and VPN 4 would essentially be the same.

FIGURE 38 Static route to CE network example



CE 1 configuration

This configuration example describes what is required to operate the CE 1 router in the diagram above. In this example, a default static route is configured between the CE 1 router and the attached interface of PE 1.

```
      device(config)# ip route 0.0.0.0/0 10.33.33.3
device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# ip address 10.33.33.2
```

CE 5 configuration

This configuration example describes what is required to operate the CE 5 router in the diagram above. In this example, a default static route is configured between the CE 5 router and the attached interface of PE 4.

```
      device(config)# ip route 0.0.0.0/0 10.33.36.3
device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# ip address 10.33.34.2
```

PE 1 configuration

This configuration example describes what is required to operate the PE 1 router in the diagram above. In this example, the VRF VPN1 is created with a unique route descriptor consisting of the BGP AS number (1) and a random other number (2), and route targets are set for import and export. The VRF (VPN1) is defined on the interface that connects to CE 1. A static route is configured between this router and the network connected to CE 1. IBGP with extended community attributes is configured between PE 1 and PE 4.

The OSPF area is specified as 0 and MPLS is set up for OSPF traffic engineering. In addition, MPLS is configured with a signaled LSP named "tunnel1" to PE 4.

```
      device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.2.2.1/24
device(config-lbif-1)# ip ospf area 0
device(config-lbif-1)# exit

device(config)# vrf VPN1
device(config-vrf-vpn1)# rd 1:1
device(config-vrf-vpn1)# route-target export 100:1
device(config-vrf-vpn1)# route-target import 100:2
device(config-vrf-vpn1)# exit-vrf

device(config)# ip route vrf VPN1 10.1.2.0/24 10.33.33.2
device(config)# router bgp
device(config-bgp)# local-as 1
device(config-bgp)# neighbor 10.2.2.2 remote-as 1
device(config-bgp)# neighbor 10.2.2.2 update-source loopback 1
device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.2.2.2 activate
device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# redistribute static
device(config-bgp-ipv4u-vrf)# exit

device(config)# router ospf
device(config-ospf-router)# area 0
device(config-ospf-router)# exit

device(config)# interface ethernet 1/2
device(config-if-e10000-1/2)# ip address 10.33.34.1/24
device(config-if-e10000-1/2)# ip ospf area 0
device(config-if-e10000-1/2)# exit

device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# traffic-engineering ospf
device(config-mpls)# mpls-interface eth 1/2
device(config-mpls)# lsp tunnel1
device(config-mpls-lsp-tunnel1)# to 10.2.2.2
device(config-mpls-lsp-tunnel1)# enable
device(config-mpls-lsp-tunnel1)# exit

device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# vrf forwarding VPN1
device(config-if-e10000-1/1)# ip address 10.33.33.3/24
device(config-if-e10000-1/1)#
```

PE 4 configuration

This configuration example describes what is required to operate the PE 4 router in the diagram above. In this example, the VRF VPN1 is created with a unique route descriptor consisting of the BGP AS number (1) and a random other number (2), and route targets are set for import and export. The VRF (VPN1) is defined on the interface that connects to CE 5. A static route is configured between this router and the network connected to CE 5.

IBGP with extended community attributes is configured between PE 1 and PE 4. The OSPF area is specified as 0 and MPLS is set up for OSPF traffic engineering. In addition, MPLS is configured with a signalled LSP named "tunnel1" to PE 1.

```
      device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.2.2.2/32
device(config-lbif-1)# ip ospf area 0
device(config-lbif-1)# exit

device(config)# vrf VPN1
device(config-vrf-vpn1)# rd 1:2
device(config-vrf-vpn1)# route-target export 100:2
device(config-vrf-vpn1)# route-target import 100:1
device(config-vrf-vpn1)# exit-vrf

device(config)# ip route vrf VPN1 10.1.2.0/24 10.33.36.2
device(config)# router bgp
device(config-bgp)# local-as 1
device(config-bgp)# neighbor 10.2.2.1 remote-as 1
device(config-bgp)# neighbor 10.2.2.1 update-source loopback 1
device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.2.2.1 activate
device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# redistribute static
device(config-bgp-ipv4u-vrf)# exit

device(config)# router ospf
device(config-ospf-router)# area 0
device(config-ospf-router)# exit

device(config)# interface ethernet 1/2
device(config-if-e10000-1/2)# ip address 10.33.35.1/24
device(config-if-e10000-1/2)# ip ospf area 0
device(config-if-e10000-1/2)# exit

device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# traffic-engineering ospf
device(config-mpls)# mpls-interface eth 1/2
device(config-mpls)# lsp tunnel1
device(config-mpls-lsp-tunnel1)# to 10.2.2.1
device(config-mpls-lsp-tunnel1)# enable
device(config-mpls-lsp-tunnel1)# exit

device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# vrf forwarding VPN1
device(config-if-e10000-1/1)# ip address 10.33.36.3/24
device(config-if-e10000-1/1)# exit
```

Configuring a static default route on a CE router

To allow route exchange between a CE router and its associated PE router, a static default route must be created to the interface on the associated PE router where the VPN is enabled. In this example, the PE 1 router has the VRF "VPN1" enabled on port 1/1, which has the IP address 10.33.33.3. To create a default static route from CE 1 to this interface on PE 1, enter the following command.

```
      device(config)# ip route 0.0.0.0 10.33.33.3
```

Configuring a static default route on a PE router

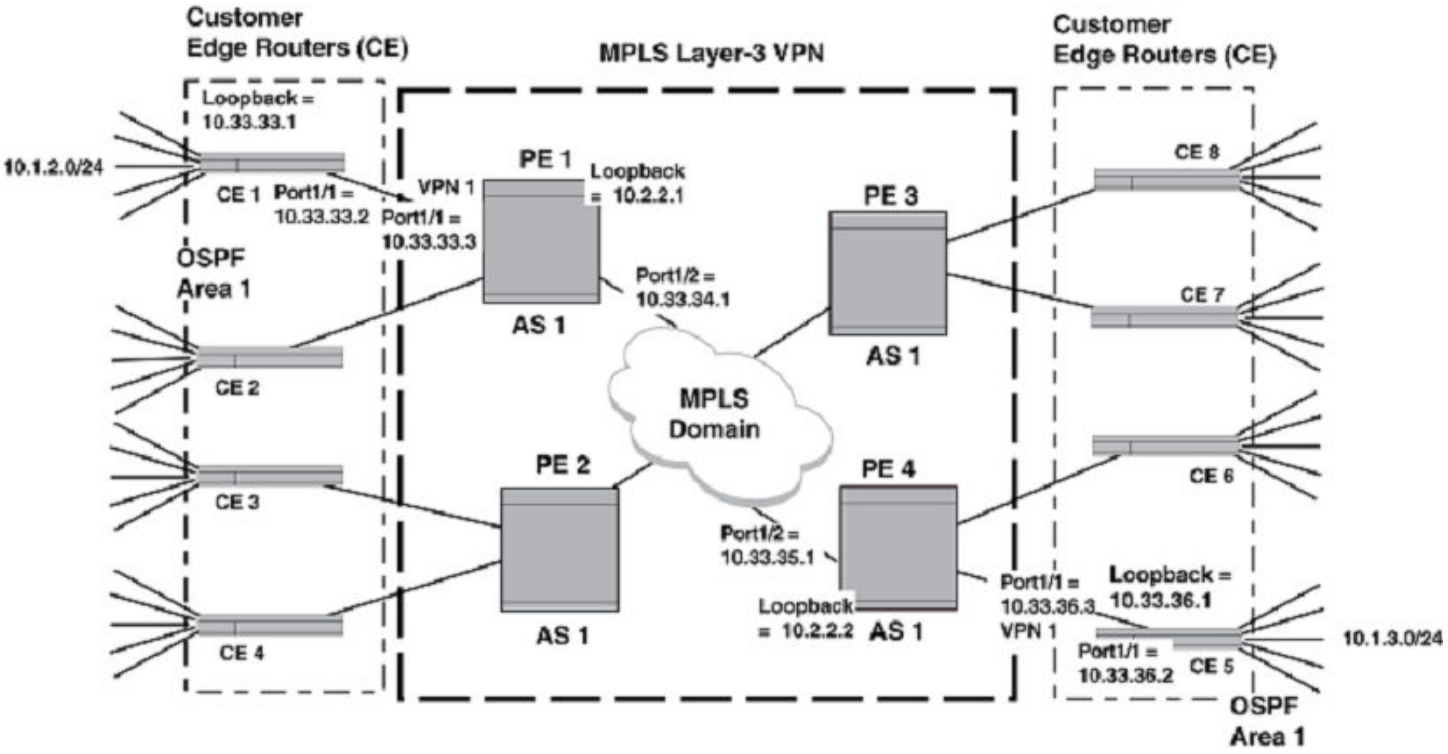
To allow route exchange between a PE router and its associated CE router, a static route must be created to the route that the user wants to provide access to with a next hop consisting of the IP address of the interface that is connected to the VRF. In this example, the IP address of the connected port on the CE router is 10.33.33.2, and the address on the CE that is provided access from the PE's VRF is 10.1.2.0/24. To create a static route from PE 1 to CE 1, enter the following command.

```
device(config)# ip route vrf VPN1 10.1.2.0/24 10.33.33.2
```

OSPF for route exchange

OSPF can be used to exchange routes between CE routers and PE routers. In this situation, OSPF must be enabled on the CE router with a local area and enabled on the interface that is connected to the interface of the PE that is associated with the VRF that the user wants to advertise OSPF routes on. On the PE router, the VRF must be enabled in BGP to redistribute OSPF routes, and OSPF must be enabled for the VRF and configured to redistribute routes from BGP-VPNv4. The diagram below provides an example of a network where OSPF is used to exchange routes between CE routers and PE routers.

FIGURE 39 OSPF to CE network example



To configure OSPF to exchange routes between PE routers and CE routers, the user must perform the configuration steps listed below.

1. Configuring OSPF on the CE router on page 301.
2. Enabling OSPF on the CE router interface on page 301.
3. Configuring the VRF on the PE router to redistribute OSPF routes on page 301.
4. Configuring OSPF on the PE router to redistribute BGP-VPNv4 routes on page 301.
5. Enabling OSPF on the PE router interface on page 301.

Configuring OSPF on the CE router

To allow OSPF route exchange between a CE router and its associated PE router, OSPF must be enabled on the CE router. To configure OSPF on the CE 1 router for local area 1 in Figure 39 on page 300 and enable it to redistribute static routes through OSPF, enter the following commands.

```
      device(config)# router ospf
device(config-ospf-router)# area 1
device(config-ospf-router)# redistribute static
```

Enabling OSPF on the CE router interface

To allow OSPF route exchange between a CE router and its associated PE router, OSPF must be enabled on the interface that connects to the VRF-enabled interface of its associated PE router. To configure OSPF on the interface of the CE 1 router in Figure 39 on page 300 that is connected to the VRF VPN1 associated interface on PE 1, enter the following commands.

```
      device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# ip ospf area 1
device(config-if-e10000-1/1)# ip address 10.33.33.2
```

Configuring the VRF on the PE router to redistribute OSPF routes

To allow OSPF route exchange between a specified VRF on a PE router and its associated CE router, the VRF must be enabled to redistribute OSPF routes. To enable the VRF VPN1 on PE 1 router in Figure 39 on page 300 to redistribute OSPF routes, enter the following commands.

```
      device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# redistribute ospf match internal
device(config-bgp-ipv4u-vrf)# redistribute ospf match external1
device(config-bgp-ipv4u-vrf)# redistribute ospf match external2
```

Configuring OSPF on the PE router to redistribute BGP-VPNv4 routes

To allow OSPF route exchange between a specified VRF on a PE router and its associated CE router, OSPF must be configured to redistribute BGP routes from the local AS. To enable OSPF on PE 1 in Figure 39 on page 300 and configure it to redistribute BGP-VPNv4 routes into OSPF, enter the following commands.

```
      device(config)# router ospf vrf VPN1
device(config-ospf-router)# domain-id 0.0.0.100
device(config-ospf-router)# domain-tag 1200
device(config-ospf-router)# area 1
device(config-ospf-router)# redistribute bgp
```

Enabling OSPF on the PE router interface

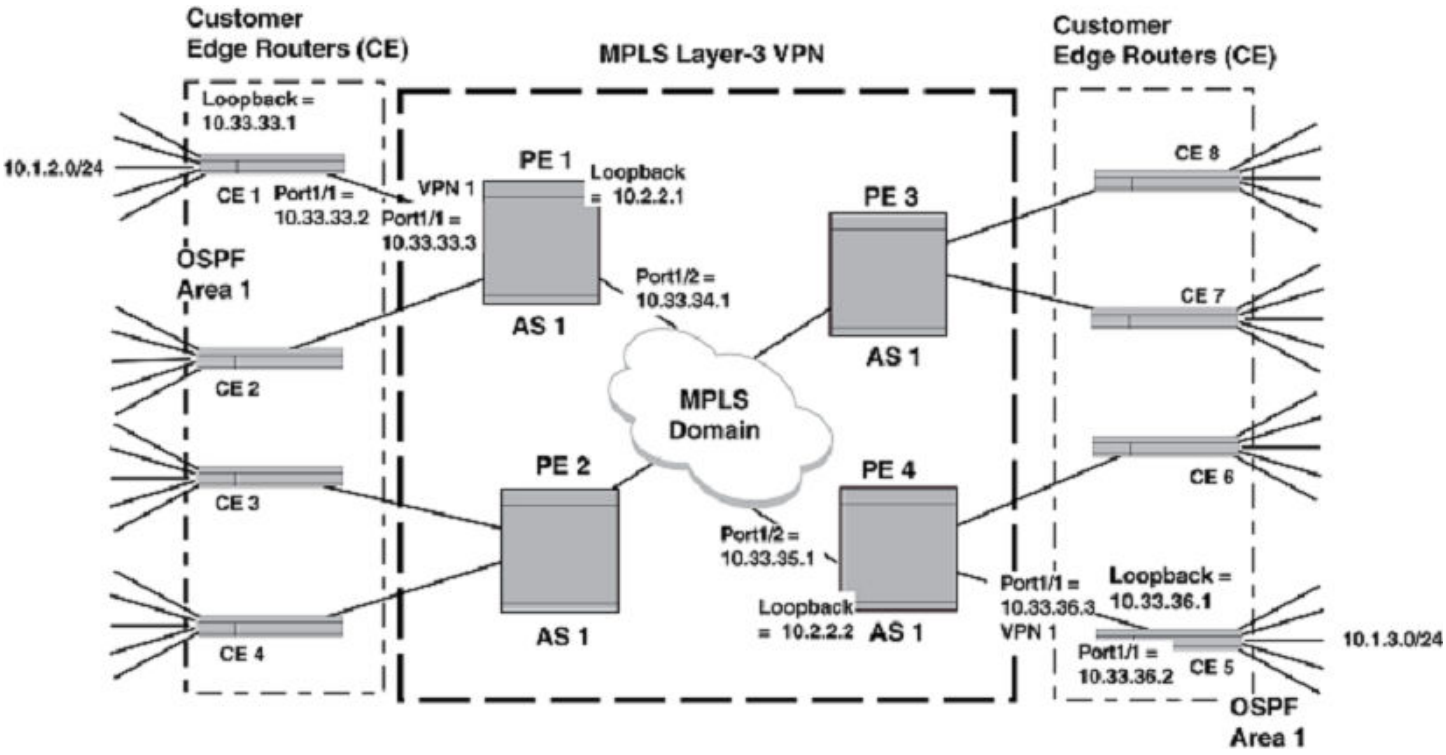
To allow OSPF route exchange between a PE router and its associated CE router, OSPF must be enabled on the PE's interface that is associated with the VRF and connected to the PE router. To configure OSPF on the interface of the PE 1 router that is associated with VRF VPN1 in OSPF for route exchange on page 300 to CE 1, enter the following commands.

```
      device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# vrf forwarding VPN1
device(config-if-e10000-1/1)# ip address 10.33.33.3/24
device(config-if-e10000-1/1)# ip ospf area 1
```

OSPF to CE configuration example

In this example, the network shown in the diagram below is configured for OSPF to forward routes between the networks attached to the CE routers and the PE routers. IBGP is used to forward routes between the PE routers and an LSP tunnel is configured across the MPLS domain. The diagram below contains all of the network addresses and AS numbers required to perform this configuration. The configurations are shown for only the CE 1, CE 5, PE 1 and PE 5 routers, which demonstrates what is required to make VPN1 work. The configurations for VPN2, VPN3, and VPN 4 would essentially be the same.

FIGURE 40 OSPF to CE network example



CE 1 configuration

This configuration example describes what is required to operate the CE 1 router in the diagram above. In this example, a static route is configured between the external network (10.1.2.0/24) and the loopback interface. OSPF is configured to redistribute static routes between the CE 1 router and the attached interface of PE 1.

```
      device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.33.33.1/32
device(config-lbif-1)# exit

device(config)# ip route 10.1.2.0/24 10.33.33.1
device(config)# router ospf
device(config-ospf-router)# area 1
device(config-ospf-router)# redistribute static
device(config-ospf-router)# exit

device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# ip address 10.33.33.2
device(config-if-e10000-1/1)# ip ospf area 1
device(config-if-e10000-1/1)# exit
```

CE 5 configuration

This configuration example describes what is required to operate the CE 5 router in the diagram above. In this example, a static route is configured between the external network (10.1.3.0/24) and the loopback interface. OSPF is configured to redistribute static routes between the CE 5 router and the attached interface of PE 4.

```
      device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.33.36.1/32
device(config-lbif-1)# exit

device(config)# ip route 10.1.3.0/24 10.33.36.1
device(config)# router ospf
device(config-ospf-router)# area 1
device(config-ospf-router)# redistribution static
device(config-ospf-router)# exit

device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# ip address 10.33.36.2
device(config-if-e10000-1/1)# ip ospf area 1
device(config-if-e10000-1/1)# exit
```

PE 1 configuration

This configuration example describes what is required to operate the PE 1 router in the diagram above. In this example, the VRF VPN1 is created with a unique route descriptor consisting of the BGP AS number (1) and a random other number (2), and route targets are set for import and export. The VRF (VPN1) is defined on the interface that connects to CE 1. OSPF is configured on the VRF named VPN1 to exchange routes with CE 1 and to redistribute routes from across the MPLS domain.

IBGP with extended community attributes is configured between PE 1 and PE 4. The OSPF area is specified as 0 and MPLS is set up for OSPF traffic engineering. In addition, MPLS is configured with a signalled LSP named "tunnel1" to PE 1.

```
      device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.2.2.1/24
device(config-lbif-1)# ip ospf area 0
device(config-lbif-1)# exit

device(config)# vrf VPN1
device(config-vrf-vpn1)# rd 1:1
device(config-vrf-vpn1)# route-target export 100:1
device(config-vrf-vpn1)# route-target import 100:2
device(config-vrf-vpn1)# exit-vrf

device(config)# router bgp
device(config-bgp)# local-as 1
device(config-bgp)# neighbor 10.2.2.2 remote-as 1
device(config-bgp)# neighbor 10.2.2.2 update-source loopback 1
device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.2.2.2 activate
device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# redistribute ospf
device(config-bgp-ipv4u-vrf)# exit

device(config)# router ospf vrf VPN1
device(config-ospf-router)# domain-id 10.0.0.100
device(config-ospf-router)# domain-tag 10.0.0.100
device(config-ospf-router)# area 1
device(config-ospf-router)# redistribute bgp
device(config-ospf-router)# exit

device(config)# router ospf
device(config-ospf-router)# area 0
device(config-ospf-router)# exit

device(config)# interface ethernet 1/2
device(config-if-e10000-1/2)# ip address 10.33.34.1/24
device(config-if-e10000-1/2)# ip ospf area 0
device(config-if-e10000-1/2)# exit

device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# traffic-engineering ospf
device(config-mpls)# mpls-interface eth 1/2
device(config-mpls)# lsp tunnel1
device(config-mpls-lsp-tunnel1)# to 10.2.2.2
device(config-mpls-lsp-tunnel1)# enable
device(config-if-e10000-1/2)# exit

device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# vrf forwarding VPN1
device(config-if-e10000-1/1)# ip address 10.33.33.3/24
device(config-if-e10000-1/1)# ip ospf area 1
device(config-if-e10000-1/1)# exit
```

PE 4 configuration

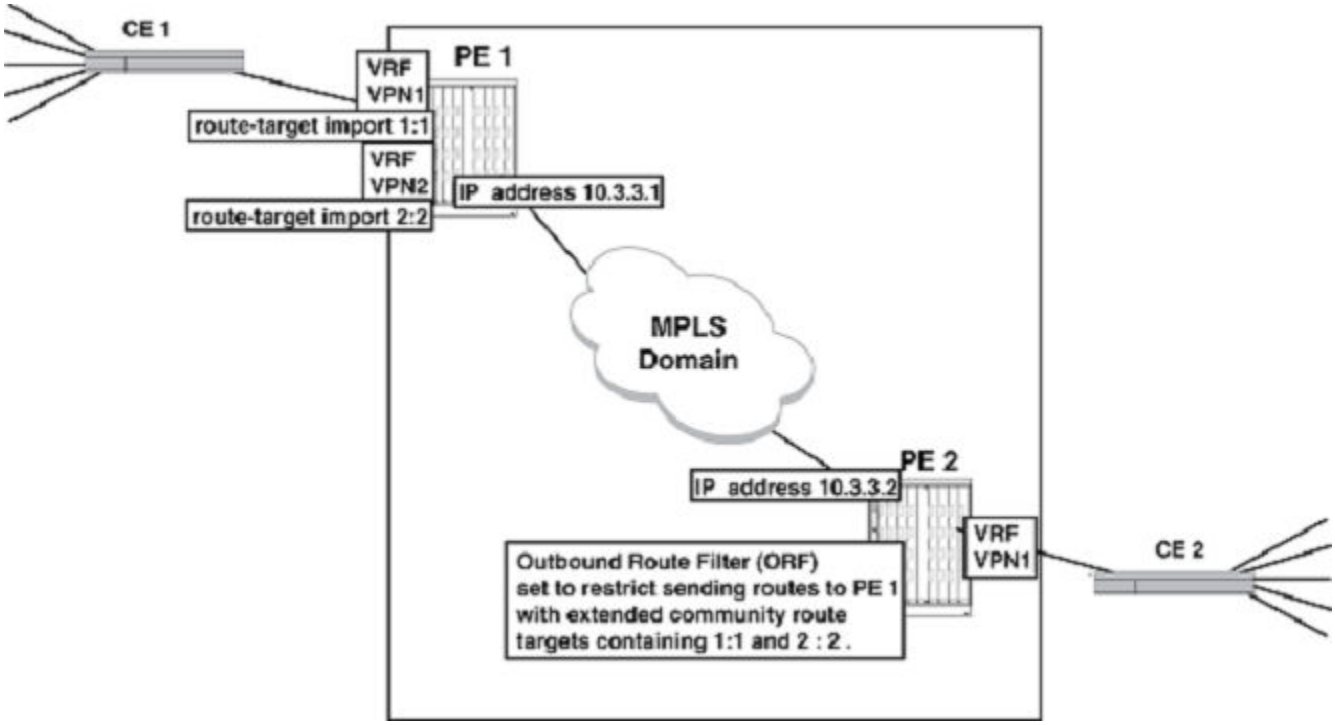
This configuration example describes what is required to operate the PE 4 router in the diagram above. In this example, the VRF VPN1 is created with a unique route descriptor consisting of the BGP AS number (1) and a random other number (2), and route targets are set for import and export. The VRF (VPN1) is defined on the interface that connects to CE 5. OSPF is configured on the VRF named VPN1 to exchange routes with CE 5 and to redistribute routes from across the MPLS domain.

```
      device(config)# interface loopback 1
device(config-lbif-1)# ip address 10.2.2.2/32
device(config-lbif-1)# ip ospf area 1
device(config-lbif-1)# exit

device(config)# vrf VPN1
device(config-vrf-vpn1)# rd 2:1
device(config-vrf-vpn1)# route-target export 100:2
device(config-vrf-vpn1)# route-target import 100:1
device(config-vrf-vpn1)# exit-vrf

device(config)# router bgp
device(config-bgp)# local-as 1
device(config-bgp)# neighbor 10.2.2.1 remote-as 1
device(config-bgp)# neighbor 10.2.2.1 update-source loopback 1
device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.2.2.1 activate
device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# redistribute ospf
device(config-bgp-ipv4u-vrf)# exit

device(config)# router ospf vrf VPN1
device(config-ospf-router)# domain-id 10.0.0.100
device(config-ospf-router)# domain-tag 10.0.0.100
device(config-ospf-router)# area 1
device(config-ospf-router)# redistribute bgp
device(config-ospf-router)# exit

device(config)# router ospf
device(config-ospf-router)# area 0
device(config-ospf-router)# exit

device(config)# interface ethernet 1/2
device(config-if-e10000-1/2)# ip ospf area 0
device(config-if-e10000-1/2)# ip address 10.33.35.1/24
device(config-if-e10000-1/2)# exit

device(config)# router mpls
device(config-mpls)# policy
device(config-mpls-policy)# traffic-engineering ospf
device(config-mpls)# mpls-interface eth 1/2
device(config-mpls)# lsp tunnel1
device(config-mpls-lsp-tunnel1)# to 10.2.2.1
device(config-mpls-lsp-tunnel1)# enable
device(config-mpls-lsp-tunnel1)# exit

device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# vrf forwarding VPN1
device(config-if-e10000-1/1)# ip address 10.33.36.3/24
device(config-if-e10000-1/1)# ip ospf area 1
device(config-if-e10000-1/1)# exit
```

Cooperative route filtering

The Cooperative Route Filtering feature allows the user to move the filtering function of a route-target import filter to a peer. In this situation, an Outbound Route Filter (ORF) is derived from the contents of all of the **route-target import** commands of BGP configured VRFs on a PE and shared with a peer PE. This ORF is then used to exclude any routes that are blocked by that ORF from being sent by the peer PE to the PE with the **route-target import** commands from which the ORF was derived. For example, in the diagram below the routes that are admitted into VPN1 and VPN2 have route targets of 1:1 and 2:2. The user can use the cooperative route filtering feature to send an ORF that is derived from the route-target import commands on PE 1 to PE 2 to only accept these routes.

FIGURE 41 Cooperative route filtering example



The following example shows the commands required to configure VRF VPN1 on PE 1 in the diagram above with an import route-target of 1:1 and VRF VPN2 on PE 1 with an import route-target import of 2:2.

```
      device(config)# vrf VPN1
device(config-vrf-VPN1)# route-target import 1:1
device(config-vrf-VPN1)# exit-vrf

device(config)# vrf VPN2
device(config-vrf-VPN2)# route-target import 2:2
device(config-vrf-VPN2)# exit-vrf
```

The following commands configure PE 1 to send the filter derived from the import route-target commands in VPN1 and VPN2 to PE 2.

```
      device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.3.3.2 capability orf extended-community send-vrf-filter
```

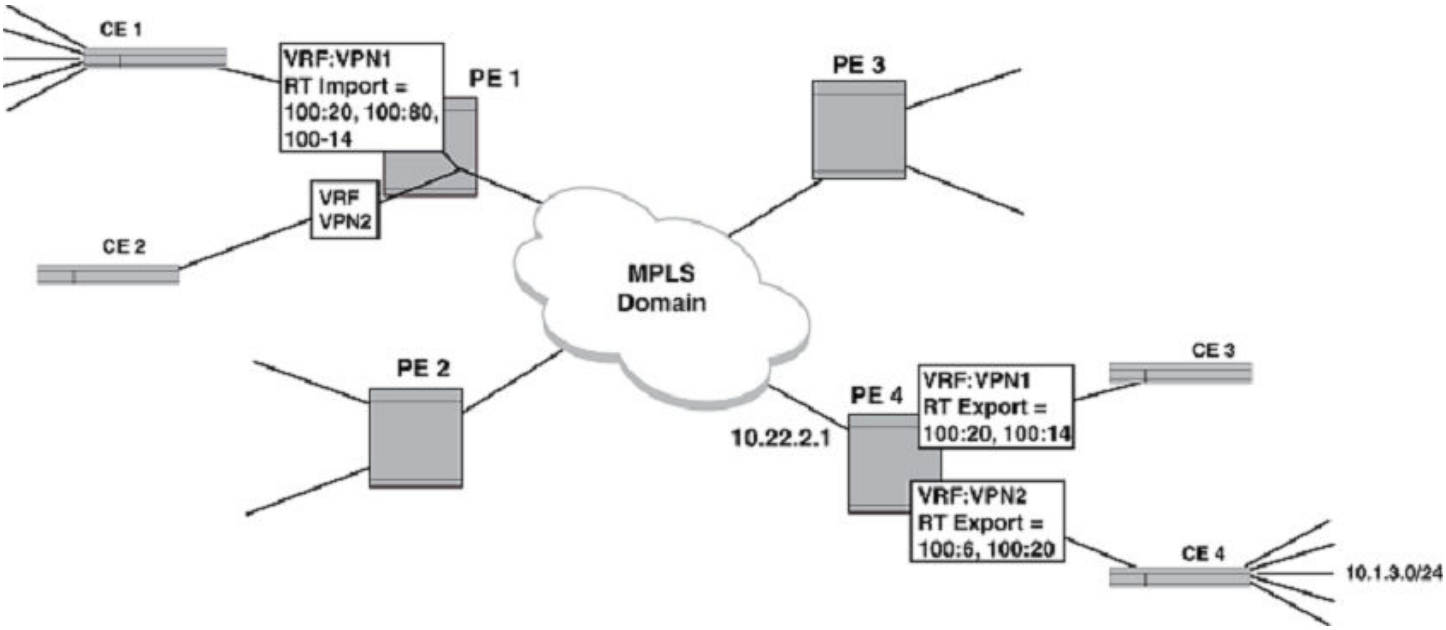
The following commands configure PE 2 to receive the filter derived from the import route-target commands in VPN1 and VPN2 on PE 1.

```
      device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-ivpnv4u)# neighbor 10.3.3.1 capability orf extended-community receive
```

Using an IP extcommunity variable with route map

In the diagram below, the VRF named "VPN1" on PE 1 is set to import routes with RT 100:14, 100:20 and 100:80. The VRF named "VPN1" on PE 4 is configured to export routes with RT 100:20 and 100:14. The VRF named "VPN2" on PE 4 is configured to export routes with RT 100:6 and 100:20. A route-map is configured from a BGP neighbor command on PE 1 to not install all routes from PE 4 with RT 100:6. This blocks all routes from VPN2 being sent to PE 1.

FIGURE 42 IP Extcommunity and route-map usage



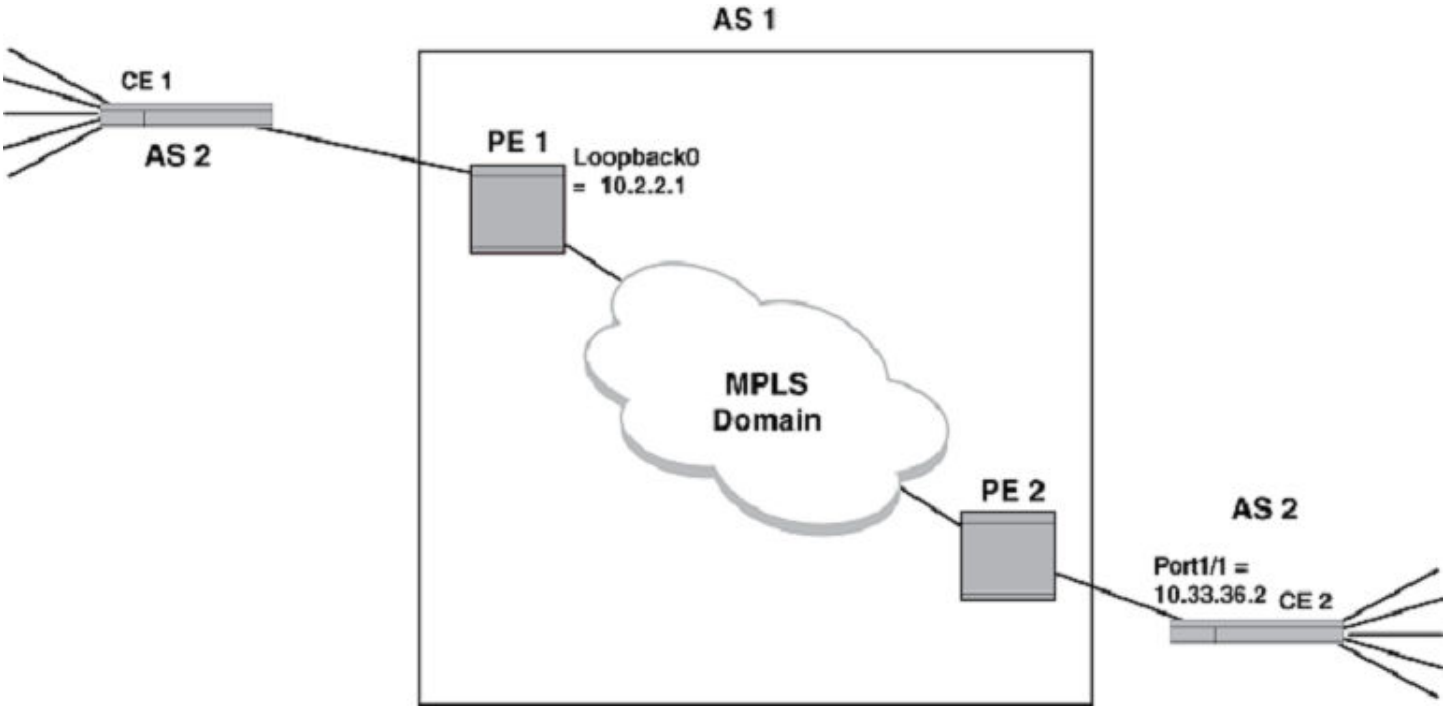
The following example shows the configuration commands required on the PE 1 router for the example shown in the diagram above. In this example, the **route-map ExcludeRoute** has an *extcommunity* value that references the extcommunity 20. The **ip extcommunity-list** command specifies that routes with RT 100:6 are to be denied. The **neighbor route-map** command exports the ExcludeRoute route-map to the BGP neighbor PE 4. Consequently, PE 4 blocks the export or route-target 100:6 to PE 1. This blocks all routes from VPN2 on PE 4 from being sent to PE 1.

```
      device(config)# router bgp
device(config-bgp)# local-as 100
device(config-bgp)# neighbor 10.22.2.1 remote-as 100
device(config-bgp)# address-family vpnv4 unicast
device(config-bgp-vpnv4u)# neighbor 10.22.2.1 activate
device(config-bgp-vpnv4u)# neighbor 10.22.2.1 route-map in ExcludeRoute
device(config-bgp-vpnv4u)# neighbor 10.22.2.1 send-community extended
device(config-bgp-vpnv4u)# exit

device(config)# route-map ExcludeRoute permit 10
device(config-routemap ExcludeRoute)# match extcommunity 20
device(config-routemap ExcludeRoute)# exit

device(config)# ip extcommunity-list 20 deny RT 100:6
device(config)# vrf VPN1
device(config-vrf-vpn1)# rd 1:1
device(config-vrf-vpn1)# route-target import 100:20
device(config-vrf-vpn1)# route-target import 100:80
device(config-vrf-vpn1)# route-target import 100:14
device(config-vrf-vpn1)# exit-vrf
```

Autonomous system number override

In the example shown in the diagram below the service providers network is in AS1 and the customer wants both of his CE routers at different sites to use AS 2. When a route is sent from CE 1 to CE 2, it contains an AS_PATH attribute containing AS 2. When CE 2 sees that the AS_PATH attribute contains its own AS number, it rejects the route.

FIGURE 43 AS number override example



One solution to this problem is to configure PE 2 to override the AS_PATH attribute that contains AS 2. When this is enabled, the PE router determines when the AS_PATH attribute in a route intended for a neighbor CE contains the same AS number as the CE. When this is determined, the PE router substitutes its own AS number for the CE's in the AS_PATH attribute. The CE is then able to receive the route. The following addition conditions apply when this feature is in effect:

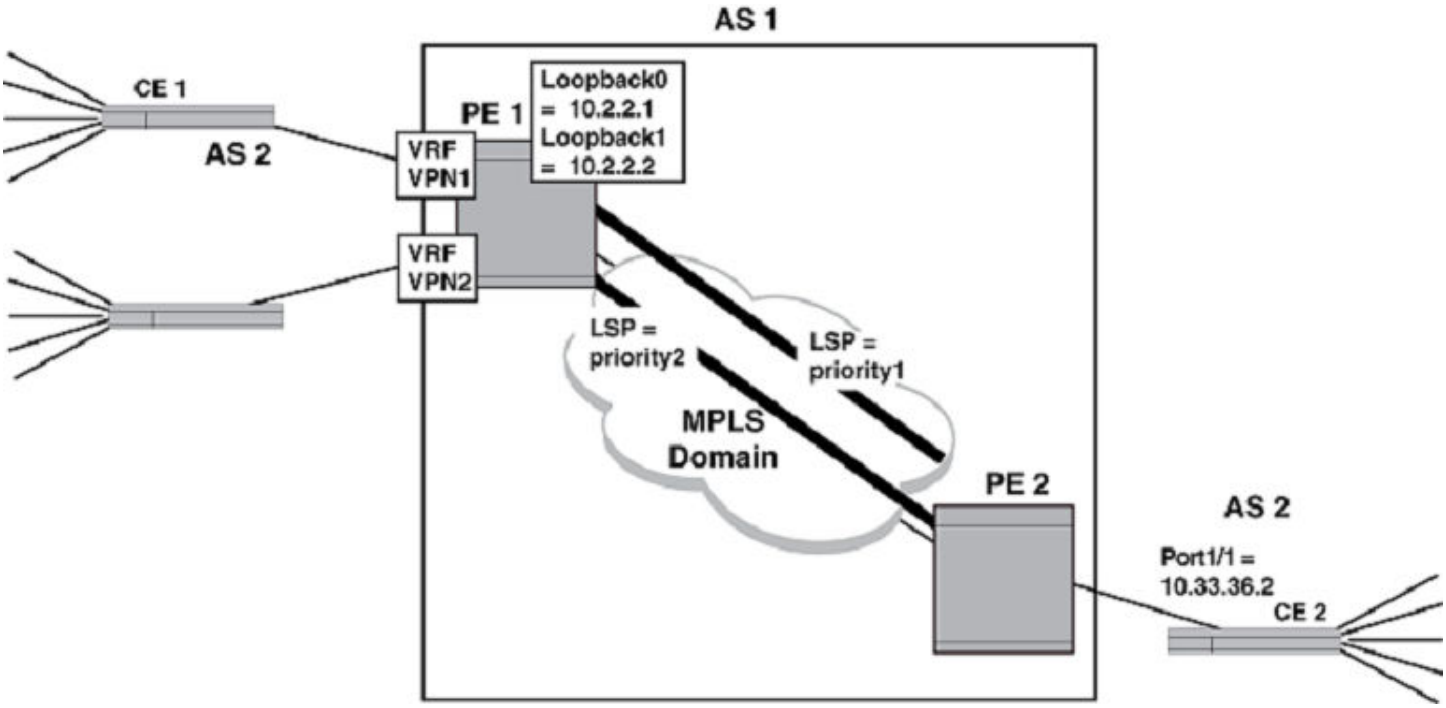
The following example describes the configuration of PE 2 required to enable Autonomous System number override for the BGP neighbor CE 2.

```
      device(config)# router bgp
device(config-bgp)# local-as 1
device(config-bgp)# neighbor 10.2.2.1 remote-as 1
device(config-bgp)# neighbor 10.2.2.1 update-source loopback0
device(config-bgp)# address-family ipv4 unicast vrf VPN1
device(config-bgp-ipv4u-vrf)# neighbor 10.33.36.2 remote-as 2
device(config-bgp-ipv4u-vrf)# neighbor 10.33.36.2 as-override
```

Setting an LSP for each VRF on a PE

The diagram below provides an example of assigning a different LSP for each VRF on a PE. In this example, PE 1 contains two VRFs: VPN1 and VPN2. It also contains two loopback interfaces with the following IP addresses: Loopback 1 = 10.2.2.1 and Loopback 2 = 10.2.2.2. Nexthop addresses for VPN1 and VPN2 can be created separately to Loopback 1 and Loopback 2. Then, different LSPs are assigned to each of the Loopback addresses.

FIGURE 44 Support per-VRF BGP nexthop



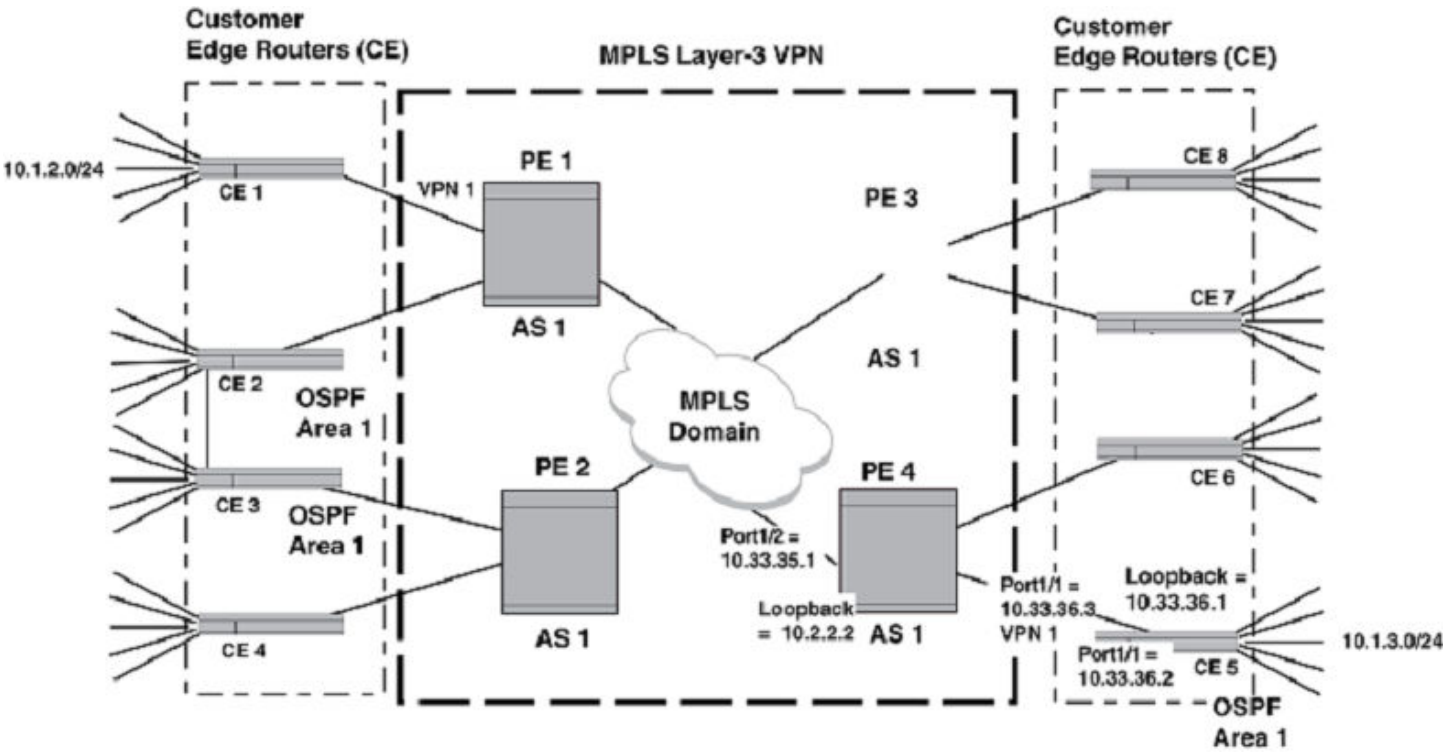
The following configuration example shows the elements in the PE 2 configuration required to make this example operate.

```
      device(config)# vrf VPN1
device(config-vrf-vpn1)# bgp next-hop loopback 1
device(config-vrf-vpn1)# exit-vrf
device(config)# vrf VPN2
device(config-vrf-vpn2)# bgp next-hop loopback 2
device(config-vrf-vpn2)# exit-vrf

device(config)# router mpls
device(config-mpls)# mpls-interface ethernet 1/1
device(config-mpls)# lsp priority1
device(config-mpls-lsp-priority1)# to 10.2.2.2
device(config-mpls-lsp-priority1)# primary-path prim-path1
device(config-mpls-lsp-priority1)# secondary-path sec-path1
device(config-mpls-lsp-priority1)# enable
device(config-mpls)# lsp priority2
device(config-mpls-lsp-priority2)# to 10.2.2.1
device(config-mpls-lsp-priority2)# primary prim-path2
device(config-mpls-lsp-priority2)# secondary sec-path2
device(config-mpls-lsp-priority2)# enable
```

OSPF sham links

In the example shown in the figure below, CE 2 and CE 3 are both in OSPF Area 1 and connect to the same service provider network through different PEs. An additional backdoor connection is configured between them over another network. OSPF recognizes the backdoor connection as an Intra-area connection and the connection through the service provider network as an Inter-network connection. Because OSPF favors Intra-area routes over Inter-network routes, most traffic between CE 2 and CE 3 travels across the backdoor link. When this is the preferred link in the network, the configuration is as it should be. However, when the user prefers traffic between the two networks to be routed across the service provider network, this configuration can causes problems.

FIGURE 45 BGP or MPLS VPN with OSPF backdoor link

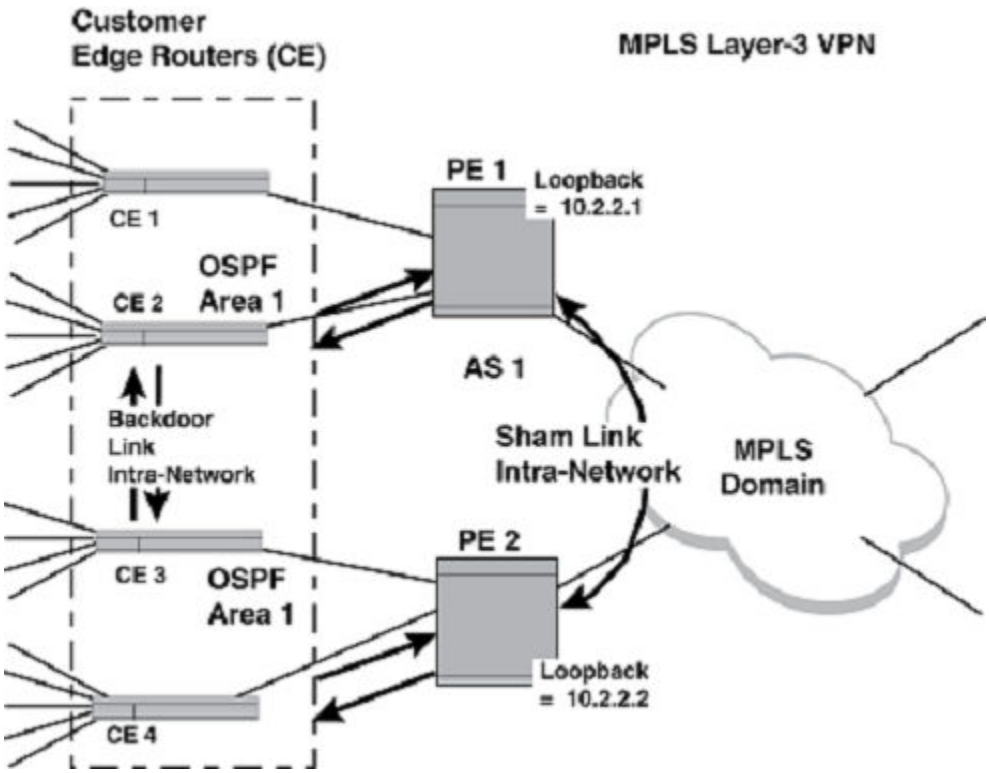


Problems can be avoided by creating a virtual intra-area OSPF link between two PEs. This virtual link is called a sham link. A sham link directs OSPF to treat the route through the service provider network as an intra-area link. A cost is assigned to the sham link to help the OSPF network determine when to route over the sham link route and when to use the backdoor link. Because this virtual link (sham-link) is an intra-area link, the OSPF areas in which each of the PEs reside must be the same.

NOTE

For sham links to work, OSPF cannot be configured on the loopback interface in the applicable area.

FIGURE 46 BGP or MPLS VPN with OSPF including Sham link and backdoor link



This configuration example describes the additional configuration required to create a sham link between PE 1 and PE 2 in the example shown in the figure above. In this example, the VRF VPN1 is added to the loopback interface configuration, and a sham link with a cost of 10 is created between the loopback interfaces on PE 1 and PE 2.

After this configuration is implemented, routes between CE 2 and CE 3 over the service provider network is preferred to the backdoor link that exists between these CEs.

PE 1 configuration

```
      device(config)# interface loopback 1
device(config-lbif-1)# vrf forwarding VPN1
device(config-lbif-1)# ip address 10.2.2.1/24
device(config)# vrf VPN1
device(config)# router ospf vrf VLAN1
device(config-ospf-router)# area 1 sham-link 10.2.2.1 10.2.2.2 cost 10
device(config-ospf-router)# redistribution bgp
```

PE 2 configuration

```
      device(config)# interface loopback 1
device(config-lbif-1)# vrf forwarding VPN1
device(config-lbif-1)# ip address 10.2.2.2/24
device(config)# vrf VPN1
device(config)# router ospf vrf VLAN1
device(config-ospf-router)# area 1 sham-link 10.2.2.2 10.2.2.1 cost 10
device(config-ospf-router)# redistribution bgp
```

PE 1 configuration

```
      device(config)# interface loopback 1
device(config-lbif-1)# vrf forwarding VPN1
device(config-lbif-1)# ip address 10.2.2.1/24
device(config)# vrf VPN1
device(config)# router ospf vrf VLAN1
device(config-ospf-router)# area 1 sham-link 10.2.2.1 10.2.2.2 cost 10
device(config-ospf-router)# redistribution bgp
```

PE 2 configuration

```
      device(config)# interface loopback 1
device(config-lbif-1)# vrf forwarding VPN1
device(config-lbif-1)# ip address 10.2.2.2/24
device(config)# vrf VPN1
device(config)# router ospf vrf VLAN1
device(config-ospf-router)# area 1 sham-link 10.2.2.2 10.2.2.1 cost 10
device(config-ospf-router)# redistribution bgp
```